AIAA FLIGHT SIMULATION TECHNOLOGIES CONFERENCE

A COLLECTION OF TECHNICAL PAPERS
MONTEREY, CALIFORNIA
AUGUST 17–19, 1987

A COLLECTION
OF
TECHNICAL PAPERS

AIAA FLIGHT SIMULATION TECHNOLOGIES CONFERENCE

August 17–19, 1987/Monterey, California

General Chairman
FRANK M. CARDULLO
State Univ. of New York

Technical Program Co-Chairmen
MARTIN FLAX
Northrop Aircraft Div.

RICHARD HEINTZMAN
ASD/ENET

Administrative Chairman
R. KOLAR
Naval Postgraduate School

TABLE OF CONTENTS

AIAA NUMBER		PAGE NUMBER
87-2299	Simulation in ADA- How Do You Get There From Here?- D.McCABE, A.LA PLANTE and S.RAMACHANDRAN	1
2300	Multi-Aircraft Simulation: Some Problems and Prospects- S.SEIDENSTICKLER	8
2301	Part-Task Simulation Techniques- A.M.SIZER	17
2302	Application of Personal Computers to Real-Time Simulation Support- W.A.RAGSDALE	24
2370	Effect of Time Delay on Manual Flight Control and Flying Qualities During in-Flight and Ground-Based Simulation- R.E.BAILEY, L.H.KNOTTS, S.J.HOROWITZ and H.L.(PAT) MALONE III	30
2371	The Effects of Time Delay and Simulation Mode on Closed-Loop Pilot/Vehicle Performance: Model Analysis and Manned Simulation Results- W.H.LEVISON and B.PAPAZIAN	39
2372	Temporal Fidelity in Aircraft Simulation Visual Systems- M.S.MERRIKEN, G.E.RICCIO and W.V.JOHNSON	50
2373	The Impact of Network Delay on Two-Ship Air-to-Air Combat Simulation- H.L.(PAT) MALONE III, S.HOROWITZ, J.A.BRUNDERMAN and H.EULENBACH	55
2434	Predictive Compensation of Visual System Time Delays- D.J.SOBISKI and F.M.CARDULLO	59
2435	Frequency Response Identification of a Computer-Generated Image Visual Simulator With and Without a Delay Compensation Scheme- W.F.JEWELL, W.F.CLEMENT and J.R.HOGUE	71
2436	The Response of Airline Pilots to Flight Simulator Motion- L.D.REID and M.A.NAHONT	77
2437	Operating Experience of a Small Six Axis Motion System Inside a Dome With a Wide Angle Visual System- A.G.BARNES	86
2498	A New Simulation Model Building Process for Use in Dynamic Systems Intergration Research- P.D.ARBUCKLE, C.S.BUTTRILL and T.A.ZEILER	96
2499	Flight Simulation of Multiple Aircraft and Helicopters Using a Single High-Performance Computer- R.M.HOWE	107
2500	Development of a Real-Time Blade Element Aeroelastic Rotor- M.E.DREIER	111
2501	Nonlinear Simulation of a Flexible Aircraft in Maneuvering Flight- C.S.BUTTRILL, T.A.ZEILER and P.D.ARBUCKLE	122
2572	Simulation in Support of Advanced Cockpit Development- P.A.LYPACZEWSKI, A.D.JONES and J.W.VOORHEES	134
2574	A Flight Simulator Evaluation of the Approach Path Parameters for MLS Curved Approaches- L.J.J.ERKELENS	141
2637	A Programmable Gaming Area- K.R.HENKE	152
	LATE PAPER	
2575	AH-64 Combat Mission Simulator Tactical System - E. DREW, G. GEORGE and S. KNIGHT	155

SIMULATION IN ADA® - HOW DO YOU GET THERE FROM HERE?*

D. McCabe
A. La Plante
S. Ramachandran**

McDonnell Douglas Helicopter Company
Mesa, Arizona

ABSTRACT

This paper addresses the issues involved in transitioning to an Ada programming environment for simulation application. The advantages and disadvantages of Ada are examined from the perspective of an engineering simulation organization. System architecture, software development environment, Ada compilers/cross-compilers and software engineering methodologies are discussed. Simulation architecture selected by McDonnell Douglas Helicopter Company and lessons learned are presented.

INTRODUCTION

Ada has been mandated as the primary programming language by the Department of Defense (DoD) for mission critical programs. This emphasis on Ada has been reflected in recent training systems initiatives also. Since new training devices are built from the ground up, design and development of new simulators are somewhat straightforward. However, aircraft companies with established simulation facilities face the difficult choice of whether and how to make the transition from a mostly assembler/FORTRAN environment to Ada. McDonnell Douglas Helicopter Company is one of the companies that has been making this difficult and expensive transition. This paper examines the technical issues that are to be considered and resolved to make a successful transition. The discussion here is from the viewpoint of an organization that has to support both engineering and training simulation activities. The major question is should the simulation organization switch over to Ada at all or maintain status quo to the software development? This requires an examination of not only the technical pros and cons of Ada software development but also of the charter for the simulation organization (engineering simulation versus training simulation) and the costs associated with the decision to be taken.

The paper first discusses the pros and cons of choosing or not choosing Ada. Problems of transition based on McDonnell Douglas Helicopter Company's background and requirements are discussed. Once the decision to choose Ada has been made, the technical issues for a successful simulation implementation are reviewed. McDonnell Douglas Helicopter Company's approach to satisfy its simulation requirements is presented. The lessons learned in achieving Ada implementation are also presented.

ADVANTAGES AND DISADVANTAGES OF ADA

The advantages of using Ada in simulation applications are not much different from those for other applications. Engineering simulation is a large software intensive activity. However, in the context of aircraft development, simulation is a medium where the technical efforts of diverse engineering organizations within the company are brought to focus and where airplane concepts are validated. It is also a tool where software eventually intended for aircraft is developed and tested. For these reasons, the advantages of Ada are even more attractive for simulation application. One such consideration is the easier portability Ada offers. Portability of code has been an elusive goal of simulation software as it is with other software applications. It has been recognized for some time that standardizing on a single

* Presented at the AIAA Flight Simulation Technologies Conference, Monterey, California, August 1987.
** Member, AIAA.
® Ada is a registered trademark of the U.S. Government, Ada Joint Program Office.

language will be a major part of making this goal a reality. At this time the Ada language is being far more tightly controlled than any other language in both the language revision aspect (there is only ONE version of the language) and the compiler implementation aspect (certified validation suites). This makes Ada a stable language to standardize on, making portable code far more feasible than with a non-regulated language. Since aircraft development has become software intensive, it is extremely important to reduce software costs. To achieve this, software must be portable between the simulator, hot bench and aircraft. With Ada mandated as the higher order programming language for aircraft development, adopting Ada as the programming language for the simulator also makes good economical sense as well.

Another advantage is that, in the long term, all programmers will use a common programming language and therefore will have a far easier time transitioning from one aircraft project to another. Ada will also permit efficient utilization of programmers since they could be moved around within the company between simulation and aircraft programs depending upon company needs.

The technical advantages of Ada are even more alluring in the context of commonality between simulators, hotbenches and aircraft. Ada encourages structured object oriented design, which closely resembles the way different systems/subsystems/components operate in an aircraft. Built into the Ada language is the construct of packages which allows a mechanism for putting all the code which describes an object and its processes into a logical unit. This package can be incorporated with other packages or subprograms thereby allowing the use of these code objects throughout a software system. It is a powerful way of letting the code reflect the objects and processes necessary to control a system in an understandable manner. Ada enforces a high degree of structure by imposing the principles of modularity, abstraction, information hiding, and localization. The interfaces are embodied in the package specification and can be totally defined prior to having to work out the algorithms associated with them. Ada decreases the possibility of having the wrong variables being passed from one software unit to another thereby increasing the understanding of the flow of the software and making maintainability and modification easier.

Ada permits depiction of parallel events in an understandable manner. Most languages address this problem by interfacing from their high level language to either operating system calls (which vary from operating system to operating system) or to assembler routines which schedule multiple programs simultaneously. In Ada this concept is addressed right in the language via the TASK construct. The scheduling of simultaneous events is no longer buried in code as a call to some routine which is written in a low-level language and one has to guess what it is doing. In Ada it is labeled as a TASK with clear rendezvous points.It is in the same language as the rest of the code. This is another invaluable advantage for understanding the code for either maintenance or modification purposes as well as emulating the aircraft hardware operation in the simulator.

However, these advantages have to be weighed against the disadvantages of using Ada. One problem area is the lack of compilers with the system dependent features described in the Ada Language Reference Manual's (LRM) Chapter 13. These features include many of the system programming capabilities which are necessary for simulation applications. These include pragmas (which provide the selection criteria for mapping an entity onto the underlying machine) such as PACK (elimination of gaps in storage areas allocated to consecutive components), INLINE (machine code insertion), and INTERFACE (calling subprograms written in another language). They also include REPRESENTATION CLAUSES (imposing certain characteristics of the mapping of an entity onto the underlying machine), ADDRESS CLAUSES (specification of a required address in storage which allow interrupts to be coded), UNCHECKED DEALLOCATION and UNCHECKED CONVERSION. Although there are plans to include these features in future validation efforts, they are not tested at this time. Therefore, one of the important criteria in considering an Ada compiler is what aspects of Chapter 13 are implemented and to what degree.

Another disadvantage with the implementation of the language is the lack of optimization in both host and target Ada compilers and cross-compilers. There are at least two reasons for this. One is that Ada embodies many aspects of computing that were previously rendered to

the realm of operating systems. There is a learning curve involved in just being able to implement these aspects in a high order language. The other is that Ada is relatively new compared to other high order languages. Enough time has not passed for optimization to have been the prime focus of the implementors.

Another disadvantage is the conspicuous lack of software engineers capable of designing software which incorporates the unique features of Ada. A structured methodology of design is mandated by these features and changing the way software is normally developed in a simulation environment can be painful, especially if there are not enough experienced personnel to guide the inexperienced programmers. This disadvantage can be alleviated if concerted efforts are introduced to train programmers in the design aspects necessary to properly use this language.

The above disadvantages will diminish in impact if not totally disappear as the implementations of the language mature and the software engineering community gains experience with it.

Despite the implementation shortcomings of Ada, there are two important considerations for adopting Ada. One is whether the simulation facility is interested in training device contracts. Based on current trends in military training system procurement, it is obvious that if a switch is not made to Ada, the company will be left behind in the training market since more and more military training device contracts require Ada and the company will not have the experience or talent base to compete. If the simulation department does not choose Ada as its standard software development language, it will be required to change existing developed code for every new version of its current standard language, whatever it may be. This is due to a lack of tight control on other languages which has been imposed on the Ada language and implemented through the validation process.

However, the transition to Ada is an expensive proposition since most of the existing software is in a language other than Ada, and usually in FORTRAN. This existing code may have to be redesigned in Ada. This cost could be very high since a software redesign rather than a simple conversion is necessary to take advantage of Ada's strengths. New computers, operating systems, and compilers may have to be bought for the implementation since existing computing systems in many cases may not have Ada development and real-time execution capability. Some existing configurations permit Ada-only implementations and not simultaneous implementations of new Ada code with existing FORTRAN software.

Putting off the switch to Ada will only make it even more expensive later since additional software will need to be eventually redone in Ada. It is also important to maintain a smooth transition to an Ada environment. Ada software conversion/development has to be achieved with an existing workforce and in an economical fashion while supporting the current simulator operation, developing software for new simulators and planning for future ones.

SIMULATION REQUIREMENTS

McDonnell Douglas Helicopter Company's Engineering and Training Simulation Department (ETSD) was faced with the issue of making a decision regarding Ada in early 1986. At that time ETSD had a full mission simulator in operation in support of a research rotorcraft program. At the same time the department was in the early stages of developing a second simulator for an advanced version of an existing rotorcraft. Plans were also being made for developing a third simulator in 1987 for this program. A major part of the simulation software on the first simulator was in FORTRAN with the rest in assembler and C.

All the simulated aircraft incorporate the latest and planned advances in combat rotorcraft technology including "glass cockpits" and a full suite of avionics, weapons, and sensors. The simulators provide full flight and mission simulation including visual and sensor simulation along with moving map systems. Since the first simulator was developed during 1984, Ada was not used. Consequently, the issue of Ada was taken up later when the second simulator program was started.

Along with the new simulators McDonnell Douglas Helicopter Company also faced the issue of interactive simulation. The aircraft could participate in joint missions in mixed roles of adversaries or friendlies. The development of a system control station for typical instructor/operator functions as well as for engineers to

obtain and analyze aircraft data also required the networking of these three simulators plus other simulators as they were developed and brought on line. The requirement for the second and third simulators gave the opportunity to examine technological alternatives in terms of both hardware and software in the light of recent computer technology and Ada implementation.

TECHNICAL ISSUES

The technical issues to be resolved included computer hardware performance and requirements, software development environment, software tools, Ada compilers/cross-compilers, and software engineering.

System architecture: One of the guidelines used in the hardware evaluation was the need to use existing minicomputers as well as a special purpose processor for high speed flight dynamics and control simulation. A modular design with minimum system upgrade costs was desired since the demand for computer power invariably increases with enhancements to aircraft capability. To facilitate effective man-machine interface between the pilot and the helicopter, minimum data transport delay between processors is required. The configuration must support implementation of a wide variety of algorithms ranging from artificial intelligence to aircraft electrical and hydraulic system. The system configuration must also permit simulation at rates up to 60 Hz. Most importantly, the hardware cost must be as low as possible. The popular criterion of cost per MIPS was used as a yardstick to determine hardware cost. A wide variety of systems were evaluated including multiprocessors, multicomputers, array processors and pipeline systems. Different vendor products within each group were also evaluated. Based on the above requirements and the implementation of Ada, it was found that distributed processing with multiprocessors offered the best cost and technical solution.

Software Development Environment: While the above resolves the target or the real-time implementation system, it is equally important to address the issue of the software development system for simulation. Developing software in Ada is different from developing software in other languages due to the many aspects of this language discussed earlier. Due to this difference, the development environment becomes an extremely important tool. The necessity to understand how different types of software modules interact with each other in a multi-tasking and generic software system drives the need for software tools which allow graphical/textual representations of design and automatic documentation, as well as appropriate compilers and cross-compilers.

During the preliminary design phase of development automatic software tools for graphical representations of data flows and module constructs such as packages, tasks, and subprograms allow the software designers a standardized way of creating their designs and an excellent method for documenting it in an understandable form. During the detailed design phase, textual representation of the algorithms and variables is more uniformly presented with the aid of Programming Design Languages (PDL) and Data Dictionaries. Many of the commercial PDLs on the market also include templates which produce them in military standard formats as well as supplying automatic metrics. As pointed out earlier, the implementors of the Ada language have been slow to implement all the features in the Language Reference Manual. It therefore becomes critical to develop the criteria for evaluating compilers and cross-compilers for simulation applications in order to avoid problems at the coding phase. Appendix F of every compiler's manual and Chapter 13 of the Language Reference Manual provide the real-time features which may be necessary for simulation applications.

Once it is realized that software tools become more of a necessity when designing in this language, the question of whether the present development system is adequate becomes very important. It requires analyzing the existing development system, software tool requirements, software development tools that are currently available, and most importantly, if they will work together. If the tools do not work together, then the question is whether the present development system for Ada design and coding should be used.

Ada compilers/cross-compilers: To be cost effective the real-time (target) system may be quite different from that of the software development system, as was the case at McDonnell Douglas Helicopter Company. In this case it is important to evaluate not only the compilers but also the cross-compilers. These are the most important software tools in the

development environment. Designing a list of criteria for necessary and desirable features in an Ada compiler/cross-compiler for a given application can make the job of selecting the compiler/cross-compiler easier. Simulation code is run in real-time and therefore criteria such as ADDRESS CLAUSES (to allow interrupt capability) and pragma INTERFACE to the target systems assembly language may be necessary. The pragma PACK and REPRESENTATION CLAUSES may be added to the list of desirable compiler criteria if transporting of large volumes of data throughout the simulation system is necessary. Criteria may have to be set as to the speed of the compilation time, the run time, or both. Valid data on Ada compilers in this area is difficult to obtain as it is fairly easy for the vendors to skew the results of their tests with the mix of code they use in it. An unbiased source of information may be obtained from the Performance Issues Working Group (PIWG) of the Association for Computing Machinery (ACM). Their data are obtained from ACM volunteers running the test suites that the PIWG develops. However, the compiler being considered may not have been tested within the same machine configuration as that being considered. The PIWG also does not do an analysis on the results of the tests; they just publish the data and let the users draw their own conclusions.

If a requirement to use a validated compiler does not exist for the application under consideration (as may be the case in an engineering simulation) one could consider a non-validated compiler for development needs. Often non-validated compilers claim faster compilation and execution times than validated compilers. But it is important to examine carefully any compiler that has not been validated or is not planned to be validated. As stated earlier, the validation suites do not test all of the system dependent features needed for simulation applications but they do assure that all aspects of the language which are not system dependent are tested. This may become vitally important for future modification or porting considerations of the developed Ada code. The compilation/execution speed advantages that may be realized initially need to be weighed against the cost of future redesigns or recoding.

Once the compiler criteria has been established, one more list needs to be established: the compromise list. What trade-offs are acceptable in both the compiler and cross-compiler? They may be related to the criteria mentioned above or the associated software tools which are included with the compiler. An automatic recompilation system may outweigh the compilation speed of a compiler, especially for very large applications. Or it may not be possible to get the bit packing desired in the cross-compiler but it does offer a source target code debugging capability. Different vendors are focusing on different aspects of their compiler systems. It is worth the effort to investigate their track records also if a vendor is promising some features which are not required now but are absolutely necessary for use in the future. (Asking for customers' names from vendors for this purpose is an accepted practice). Care should be exercised in the compromise as it is emphasised again that the compiler and cross-compiler are the most important software tools in the simulation development system.

Software Engineering: Software engineering is the term applied to the activity of creating software in a disciplined and consistent way. Ada's rich set of constructs and capabilities give the software engineers the ability to create a software solution which maps more accurately the problem domain it is addressing than other higher order languages which incorporate operating system calls or low-level assembler routines to effect the same solution. Simulation tackles complex systems and this fact coupled with the enhanced capabilities Ada offers requires a well thought out methodology for designing simulation software.

There are several software development methodologies available. Few, if any, cover all the aspects of designing software from requirements through testing. Most of the software design methods embodied in these methodologies are either top-down structured, data-structure, or object-oriented design. The object-oriented method seems to be emerging as a favorite although most successful projects seem to be employing a combination of methods.

Another aspect to designing simulation software in Ada is what to do with all the developed FORTRAN code. Three options are possible: incorporate it, convert it, or re-design it in Ada. The first option is dependent on the Ada compiler one chooses. If it supports the pragma INTERFACE to the FORTRAN language, it is possible to use most of that code with minimal rewrites. The second option is the least desirable

of the three. It implies purchasing a FORTRAN to Ada software conversion tool. There are a number of them in the commercial market. However, the Ada code which emerges would neither resemble well designed Ada modules nor understandable FORTRAN. The last option is the most desirable since the final product is a simulation code in one language. It may, however,also be the most time consuming and expensive and may not be feasible initially. Whatever option is chosen, incorporation of FORTRAN code needs to be taken into account in the design methodology.

SIMULATION CONFIGURATION

Figure 1 shows the McDonnell Douglas Helicopter Company multi-ship configuration arrived at after reviewing all the issues discussed earlier. As mentioned earlier, the approach was to use a distributed processing system for real-time simulation. The configuration uses existing processors and FORTRAN and other code as they existed. All computing expansion is achieved through microprocessors thus permitting a low cost and very affordable solution. This configuration also permits the main goal of developing all new software in Ada via a development system which is cross-compiled to the target system. An existing super-mini computer with compiler and cross-compiler provide the main software development capabilities. The configuration also permits gradual redesign of FORTRAN code to Ada and phasing in of processors to execute the redesign code. PDLs, automatic document generator, and language sensitive editor are the primary software tools. These allow the development configuration to support high software productivity. Most importantly it forces a modular software/ hardware approach in the long run.

LESSONS LEARNED

The McDonnell Douglas Helicopter Company simulation department has successfully moved into developing software in Ada, but this was not achieved without some difficulties. There were a number of lessons learned along the way.

Due to the fact that many Ada compiler and cross-compiler implementors are not quite "there" with all the features necessary for simulation software it is important to decide whether Ada will be the simulation language before investment is made into software and hardware for both the development and target systems. If money is invested in target systems before investigating the cross-compilers and debug tools, the advantages expected from a faster and more economical CPU may well be nullified by the fact that there are few, if any, good cross-compilers for the target system that has been purchased. As stated previously, development systems need enhanced capabilities when developing with Ada. Vendors who supply the software packages which allow this enhancement have not made these packages for every operating system. The development system and the software packages have to be considered as a whole and not in piece-meal. It is nearly impossible to retrofit software packages to any existing machine.

Designing software in Ada needs more training than that which is offered by the compiler vendors as part of the product they sell. Just knowing the syntax of the language is not enough. As stated earlier, a methodology for design does need to accompany the learning of Ada. Otherwise the price paid is creation of non-reuseable code if design issues are ignored in developing simulation Ada software. Unfortunately just selecting a methodology may not be enough. The methodology chosen may not fit the type of software that is being developed. One approach to this dilemma is to evaluate the PDL or code which is being produced by this methodology as soon as possible. If it is too complex, try another methodology or modify the existing one.

In the area of compilers and cross-compilers, a number of lessons were learned. On the issue of a validated versus a non-validated compiler, there is no question - use a validated compiler! This does not guarantee all the capabilities needed for simulation applications but it does mean the compiler has been extensively tested by validation suites and there will be fewer problems than with a non-validated compiler. The next lesson is that validation by itself is not enough. The buyer must be aware of what implementation dependent features of the LRM is needed for their applications. Knowing Chapter 13 and being able to understand Appendix F of a compiler vendor's manual is very helpful. The final lesson is to determine what compromises are acceptable, since it may not be possible to get all the features desired in a compiler. In the same vein, it is helpful to consider which vendor is making progress in the areas of

capability we require, even if they do not have these capablities now.

CONCLUSION

This paper has reviewed the issues in transitioning to the Ada programming language for simulation applications. The issues were discussed in the light of McDonnell Douglas Helicopter Company's experience in achieving this transition. The advantages and disadvantages of Ada were examined. Technical issues such as hardware performance and requirements, software development environment, software tools, Ada compilers and cross-compilers, and software engineering were considered. McDonnell Douglas Helicopter Company's simulation configuration was discussed and lessons learned were presented. It was pointed out that the most important aspect is that all these issues must be examined in totality before any commitment is made to purchase hardware or software.

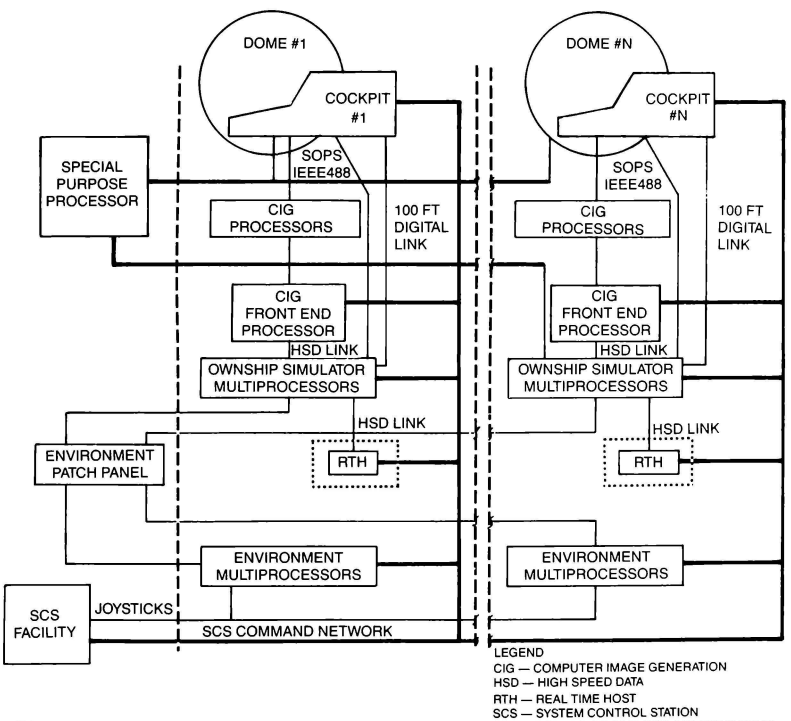


Figure 1. Simulation Architecture

MULTI-AIRCRAFT SIMULATION: SOME PROBLEMS AND PROSPECTS

Steve Seidensticker

Logicon, Inc.
Tactical and Training Systems Division
San Diego, California

Abstract

Increased dependence on flight simulation for training and tactics development demands a practical simulation in which a number of aircraft can interact. The technical challenge is formidable, but new developments in computers and digital communications are available to meet such a challenge. Cost pressures, coordination, and the requirement of agencies with differing interests to cooperate present an even greater challenge.

This paper will:

Examine some of the potential of multi-aircraft simulation such as formation flying, mutual support, and many vs. many engagements.

Take a look at some of the challenges facing such development such as inter-simulator communication, interface standards, and geographic separation constraints.

Briefly examine several programs whose goals include multi-aircraft simulation.

The Need

One of the underlying goals of flight simulation has been has been ever increasing fidelity to the "real" world. We see this in aero packages that seek to more accurately emulate the prototype aircraft in all parts of the flight envelop. We see it in image generation technology as its developers seek to show "the leaves on the trees." One area of fidelity, that is just now receiving attention, is the simulation of the entire environment in which an aircraft operates. One of the key elements of that environment is other aircraft.

More Realistic Environment for Training

Although there are times when military aircraft operate in isolation, those times are the exception, not the rule. Every military pilot training program teaches formation flying early in the program. For fighter pilots, formation flying becomes an integral part of tactics. Fighter escort and bomber pilots interact in a role of mutual support. They both learn to deal with manned interceptors in an adversary roll. Attack helicopter pilots interact with spotters and battle scene commanders in other aircraft. Emerging tactics are pitting helicopters against other helicopters in an air-to-air role. Many military pilots must also learn the skills of aerial refueling.

Tactics Development and Evaluation

An area of flight simulation equal in importance to training is an emerging role in tactics development and evaluation. This role will grow as the cost of new aircraft development continues to rise. The cost of a new aircraft is so great today that it is not possible to develop tactics to match the aircraft's capabilities. Instead, tactics must be developed to accomplish a particular mission or deal with a particular threat and the new aircraft must be matched to those tactics. This can only be done via simulation, and that simulation must include the whole environ-

ment in which a new aircraft is expected to operate.

The best example of this is the ATF program, in which all the major airframe bidders are developing simulation laboratories in which a number of simulators are interconnected so that each is part of the others' environment.

What Multi-Aircraft Simulation Can Do

Formation flying

Flying near another aircraft without contacting it is one of the basic skills taught all military pilots. It is primarily a hand-eye coordination task with communication and other potential distractions added. It is a skill achieved primarily through practice. Simulators are potentially well able to provide this practice, but only a few have the capability to do so and those provide only a prerecorded or static reference aircraft image. A true multi-aircraft simulator (all cockpits manned) would provide interaction, unpredictability, and a much more realistic environment.

Mutual Support

Almost all military flying involves mutual support in one form or another. Fighters have wing men that play very specific roles in air-to-air tactics. Bombers, EW aircraft, and escorts have very intricate mutual support roles that must be played properly to successfully press an attack and return safely. Attack helicopters, scouts, and airborne battle scene commanders interact in very complex scenarios to carry out their missions.

Training in these mutual support roles is done primarily in the classroom, where the theory, the tactics, and the roles are explained. Mutual support is practiced primarily in large, live, coordinated exercises. Because of the cost and restrictions on the use of weapons and sensors these large exercises are infrequent and their realism is compromised.

Multi-aircraft simulators will play a vital role between the classroom theory and large live exercises.

Many vs. Many Engagement

Many of the scenarios predicted for future wars involve large opposing forces, particularly in Europe. Air-to-air tactics are planned with this in mind, but there is now no cost effective way to practice these tactics. The Navy Strike Warfare Center(NSWC) and Red Flag were created, in part, to fulfil this need. Successful as they are, they can only train a relatively small fraction of fighter pilots and the realism of the exercises is compromised by restrictions on the use of sensors and weapons.

Multi-aircraft simulators will bring many NSWC/Red Flag benefits to a greater number of pilots and will raise the skill level of pilots entering these programs.

Some Basic Technical Approaches

There are but two basic conceptual approaches to building multi-aircraft simulators, single systems that employ multiple cockpits and the linking of conventional, single cockpit simulators. Multi-cockpit simulators are treated briefly in the following section, but the remainder of the paper is devoted to the linked simulator approach because the author expects that approach to dominate development.

Multi-Cockpit Simulators

Both the Air Force and the Navy have invested considerable resources in this approach. The most well know of these simulators is the Simulator for Air-to-Air Combat (SAAC) at Luke AFB. It was originally built over ten years ago but has just recently been upgraded. The Navy has developed device 2E6 (F-14 front seat) and several F/A-18 multi-cockpit simulators.

These devices are used both for training and research. Although the technology of the various subsystems differ, the devices have similar capabilities. Each has:

Two manned cockpits.

Two very wide field-of-view visual systems.

Additional simulated aircraft that can be flown by an instructor/operator or are piloted by software models.

Centralized computation system.

Centralized instructor/operator control station.

All of these devices are devoted primarily to simulating close (within visual range) air-to-air combat. While impressive in these capabilities, they are also limited to them. That is, they cannot support the other multi-aircraft simulation roles mentioned. These limitations, along with their high cost, leads this author to the conclusion that further development of dedicated two-cockpit simulators is not likely.

Linked Conventional Simulators

The technical alternative is to develop a mechanism that will permit conventional, single aircraft simulators to be linked and become part of each other's environment. This approach has a number of obvious potential advantages:

More of the total environment can be simulated. That is, the number of simulators operating together is not necessarily limited to two.

The roles of simulators so linked would not be limited to air combat maneuvering. The roles would be that of the individual simulators. The roles of linked simulators would not have to be homogeneous. It is conceivable, for example, that a WST for a strategic bomber and one for an interceptor could be linked and that crew of each could interact in practicing their respective missions.

The system would be very flexible. The simulators could be linked in a variety of ways to accomplish different tasks. Any, or all, could remain unlinked.

The simulators need not be physically adjacent. The links can be via phone lines and/or communication satellites (geographic separation does impose additional technical requirements and restrictions mentioned later).

Technical Challenges/Potential Solutions

Attractive as a linked simulator approach might be, there are a number of technical problems that must be dealt with effectively before the approach will find wide use.

Data Exchange

In modern aircraft information is exchanged via voice and data links. In a linked simulator system, information concerning the state of the simulated aircraft must also be exchanged.

State Vectors of Simulated Aircraft. The most obvious of these are position, attitude, direction, and velocities in different axes. It is also helpful to have accelerations in the different axes and other state information such as rate of turn. In order to completely describe an aircraft's contribution to others' environment one must add emitter information and weapons activity. It takes about 40 bytes of information to fully describe an aircraft's state.[1]

Tactical Data Link Information. An increasingly important aspect of multi-unit simulation is provide unit-to-unit tactical data link information. The most modern tactical data link in service, JTIDS, can, under ideal conditions, carry nearly 60K bits/sec.

Voice. Voice radio between linked simulators poses a problem due to the high bandwidth required to carry voice. It is possible to set up separate analog links for voice communication, but that imposes additional complexity. A more attractive approach is to digitize voice and carry it on the same link as the aircraft state and tactical data. In order to achieve ordinary phone line fidelity, commercial phone companies use a 64K bits/sec channel for digital voice. Newer modulation and sampling techniques are becoming avail-

able that permit equivalent fidelity at 32K bits/sec. Lower fidelity, but still intelligible voice can be carried at significantly lower bandwidths. Logicon uses a 2.4K bits/sec voice link that appears to be adequate, but which users complain "sounds like Donald Duck".

Encryption. The data exchanged in a warfare simulation between units will have to be encrypted when they reveal tactics and threat assessments. This includes tactical voice communication.

Dealing with Varying Levels of Fidelity

Benefits from linking simulators does not require that they be of the same fidelity. Therefore, any universal mechanism or standard associated with linking simulators must deal gracefully with differing levels of fidelity in the simulators linked.

Low fidelity simulators must be able to ignore information that they cannot use.

High fidelity simulators must be able to use information when it is available and must make assumptions about the same information when it is not available.

Ideally there should be no need for one simulator to have explicit knowledge of the capabilities of simulators linked to it.

Image Generation Requirements

Normally simulators will become part of each other's environment via the visual systems and/or simulated sensors. In either case, the image generators producing the images (visual, radar landmass, infrared) are the key. That is, the image generators must be able to create images of other vehicles in the environment. Currently only the top-of-the-line systems have that capability, and these systems can only show a limited number of external vehicles. There are indications that these capabilities are moving to lower priced systems.

Additional Processing Requirements

In general there are trade-offs between processing and other requirements. That is, by making systems smarter one can reduce requirements in other areas. For example, if one simulator has the capability to predict the position of another simulated vehicle based on that other vehicle's velocity and direction, there is no need to periodically communicate the other vehicle's position. Only changes to the other vehicle's velocity and direction need be communicated (this technique is expanded upon later).

The trade of increased processing requirements in exchange for lower requirements in other areas is generally attractive as processors become ever more powerful and cheaper. Better software engineering techniques and increasing programmer productivity make this trade even more attractive.

Bandwidth Limitations

The number of simulated aircraft in a single environment and the fidelity of the environment as a whole are ultimately governed by the communications that link them. The principle links that can be used in a linked simulator scheme are described below.

Local Area Networks (LAN). These generally have a high bandwidth (in some cases up to 80M bits/sec). The most popular is Ethernet (10M bits/sec). As is generally the case with communications schemes the high bandwidth is traded for distance. LANs generally can link components within one or two kilometers. This is adequate to link simulators in the same building or complex of buildings.

Land Lines. These are essentially phone lines and microwave links, either voice grade or dedicated. Voice grade lines can carry, with currently available technology, 9.6K bits/sec. Dedicated or leased lines are designed to handle as much as 1.5M bits/sec. Land lines are primarily attractive for linking simulators on the same base or on bases in the same geographic area.

Satellite communication. Satellite communication channels generally have as great

or greater bandwidth than land lines and are attractive for linking geographically separated simulators. They bring with them the problem of lengthened response time caused by the round trip delay of the signal to the satellite. This issue is discussed in the section on transport delay.

Packet-Switched Networks. These are attractive primarily because of convenience. The user needs only to make a simple connection to the network at each point of interest and the communication organization ensures that the data reaches its destination intact and in order.

These networks general use a combination of land lines and satellites. These combinations may change dynamically to accommodate other traffic on the network. This generally causes a variance in data arrival time over which the user has no control. This limits the use of long haul networks to function testing or other situations where delay of data has little impact.

Transport Delay

In any interactive system the delay between input and reaction to that input is important. If the system is intended to operate in "real-time", this delay becomes critical. With flight simulators excessive delay in generation and transfer of data between major components within the simulator has become a major issue. It is suspected in problems of poor simulation and in a phenomenon known as simulator sickness.

The interaction between simulated aircraft is not as closely coupled as the interaction between major components of a single simulator (e.g. stick/rudder inputs and visual system outputs). For this reason delays in moving information between simulators over a link is probably more tolerable. How tolerable depends on a number of factors and there are a number of techniques available to either reduce the delay or compensate for it. These issues are discussed in the following sections.

Within Immediate Geographic Location. If the simulators to be linked are connected via a LAN, the transport delay problem is negligible. In most LANs information can traverse the net itself in less than 20 ms. In experiments that Logicon has done to quantify Ethernet performance[2], the delay ranged from less than 1 ms under ideal conditions to 20 ms under worst conditions. These figures do not represent a significant transport delay when contrasted to typical delays within simulators of 100 ms.

Aggravations of Geographic Separation. Any attempt to link geographically separated simulators brings with it significant delay problems. These are caused primarily by the physics of signal propagation.

RF energy travels at 187,000 miles/sec. Electrical pulses travel through wire at about half that speed depending on the signal and the type of conductor. These are hard limits. They are not going to change, at least not with our present understanding of the universe.

Communications satellites are located in geosynchronous orbits about 23,000 miles above the equator. Every signal must must make at least that round trip which gives it an inherent delay of 245 ms. Added to that, of course, is encryption of the data, transmission via land line to the satellite transmitter, retransmission to the satellite, retransmission to the ground, retransmission via land line to the final destination, and decryption of the data. It is not possible to easily quantify these additional delays, but they can easily be equal to the inherent delay.

A signal which travels exclusively via land line theoretically has a significantly shorter route to travel. However, its propagation is slower, and unless it has a dedicated line from origin to destination, it is subject to numerous relays and circuitous routing. For long distances it is not certain that land lines have more or less delay. For short distances land lines should have a significant delay advantage.

While writing this paper, the author attempted to measure the delay by signing onto a

computer in Boston from a terminal in San Diego via a commercial long haul network (Tymnet). Response to input at the keyboard averaged about one second. It is not known whether the signal was all via land line or if it also included satellite relay.

Dealing with the Delay Problem. The delay problem will be most felt where actions and reactions are most noticed. Formation flying or very close air combat maneuvering are the most likely problem areas. There are several potential ways to alleviate the delay problem.

Judicious operational constraints can be set up so that geographically separated simulators do not attempt formation flying or close air combat. This may restrict flexibility to some extent but should not a major impact on the basic usefulness of linked simulators.

Another way to potentially beat the delay problem is to predict the state of other aircraft as it would be at the end of the delay. Such predictions must be accurate for about a second.

The most basic prediction is simple extrapolation of position based on velocity and direction. Accelerations, rates of turn, and other information can be used to refine the prediction.

It is not certain that sufficiently accurate predictions can be done for a high performance aircraft with a human at the controls. For lower performance aircraft or missiles this may be a viable approach.

Interaction of Weapons

When one vehicle fires a ballistic weapon or launches a guided missile at another, some special considerations must be made. Who makes the hit/miss determination? Who simulates the flight of the weapon? Personnel designing the DARPA SIMNET program (described later) decided to let the firing vehicle simulator make the hit determination for ballistic weapons and the target vehicle makes it for guided weapons. Each vehicle simulator displays the cues of the weapons flight from its own perspective. This approach reduces the amount of communications required to launch notification and hit/miss notification. It also negates the effect of delays if the simulators are geographically separated.

Administrative Challenges/Potential Solutions

Technical challenges are not the only ones to be faced in a program such as this. Coordination, the setting of standards, the enforcement of them are challenges of equal magnitude. Considering the record of agreement and cooperation between government agencies, and between competing elements of the simulator industry, the technical issues may be the least troublesome.

The Requirement for Standards

If there is ever to be any general mechanism to link simulators there must be a comprehensive and enforceable standard at its foundation.

Before considering the role of an underlying standard, one must consider what technical standards are and what they are not. There is a frequent misconception that to build something to a standard, one must build a functional or physical duplicate of the standard device. Sometimes that is the case, but standards are much more than that.

Restrictive Standards. A problem facing the designer of standards for a new application is how to create and apply standards that will standardize that which needs to be standard without making the standards restrictive. Restrictive standards tend to inhibit technical development and innovation. Occasionally the technical development occurs anyway and, if it is outside the standard, the standard itself must be abandoned.

Logical vs. Physical. In determining standards one must look at different aspects of the problem. Two of the most basic aspects are the logical (functional) and the physical. This is best illustrated by a look at the present international phone system. A well defined set of physical standards permit virtually any two telephones in the world to be

connected. But if the users of any two phones do not speak the same language, that is, if they do not have a logical standard, no communication can take place.

The following table illustrates some of the issues that must be considered in any standard simulator linking scheme.

Linked Simulator Areas of Standardization	
Logical	**Physical**
Data Formats	Cables
Function Assignment	Connectors
Prediction Algorithms	Signal Definition
Encryption Method	Data Rates
Control Mechanism	

Where Standards Need to be Applied

Common Position Reference. Any linked simulator scheme must, of course, have a common reference for position of all the vehicles in the environment. In older and less sophisticated simulators position is arbitrary. Current simulators that employ gaming areas that represent actual areas of the world use latitude/longitude. That standard should suffice for linked simulators as well.

Simulator/Problem Setup and Initial Conditions. A standard method must be set up for representing the initial conditions that all simulated vehicles need to assume at the start of a linked session.

Problem Control. Some standard method must be established to control a linked session. In addition to the usual run, freeze, reset, and terminate commands, the mechanism must permit individual simulators to join or withdraw from the session. Ideally, the mechanism should permit individual simulators to join or withdraw without explicit action by a central authority. A mechanism to limit the number of simulators is also probably necessary.

Representation of Aircraft State. Most simulators use a similar set of parameters to describe an aircraft's state, but there is no standard for the representation of those parameters. Such a standard is necessary for transfer of state information. A few suggestions follow:

Scaled Fixed Point vs. Floating Point. With the advent of floating point processors most programmers abandoned scaled fixed point notation. Floating point is more convenient in that it does not require the programmer to keep track of range and precision. The problem is that floating point notation is not standard among computer manufacturers. About ten years ago the IEEE published a standard for floating point notation and all instruction sets designed since that time have adopted it. But older instruction sets, still used by popular minicomputer manufacturers such as Gould, Concurrent (formerly Perkin-Elmer), Digital, and Data General use formats different from the IEEE's.

It would be possible to specify the IEEE format as the standard for inter simulator transfer and require the computer(s) in each simulator to convert the information as necessary. A more attractive solution, at least from this author's viewpoint, is to represent all state information in signed integer format, which is standard among all computer manufacturers. The scaling could be handled by representing all values in metric notation.

BAMS for Angular Information. A great deal of information representing an aircraft's state is angular (e.g. roll, pitch, yaw, rates for each, accelerations for each, lat/long, target angles). In most simulators angular information is represented in radians in floating point notation. This of course raises the problem problem mentioned above.

Prior to the popularity of floating point processors an angle was represented in a notation called a BAM (Binary Angular Measure). In BAM notation angular values can be stated with arbitrary precision (e.g. lat/long expressed as a pair of 32-bit BAMs can resolve position to about 1/3 inch at the equator). This notation also permits the manipulation of angular values with universal twos-complement integer arithmetic.

This author suggests a return to BAMs to represent angular information in any inter-simulator communication standard.

Standard Prediction/Extrapolation Algorithms. As mentioned earlier, the required bandwidth of inter-simulator communications can be greatly reduced if one only transmits changed data. This can be taken a step further in exchanging position information by extrapolating position from velocity and heading information, a technique known as dead reckoning (DR). The SIMNET program uses a scheme in which each simulated vehicle DRs the position of all vehicles in the environment, including itself. When an individual vehicle's DR position differs from the position determined by the main simulation program, a new position is broadcast to all other vehicles.

If such a scheme is used in a linked simulator mechanism, the DR algorithm must be the same in all simulators and therefore must be part of any governing standard.

Standardized Emitter Information. If linked simulators are to accurately handle Electronic Warfare/Combat they must exchange emitter information. This must include emitter identification, transmitted signal strength, frequency, pulse repetition rate, antenna pointing information, and the like. This information must be part of the standard that represents aircraft state.

Encryption Mechanism. The data that flows between operating simulators can carry information revealing tactics and aircraft and system capabilities. Any linked and geographically separated simulators will likely require an encryption mechanism to secure the data path between them. This encryption mechanism will have to be part of the standard.

The Outlook for Standards in this Area

Standards can be difficult to establish. Good standards require careful drafting and frequent review. There must be an organization behind them. Several organizations exist that could support the establishment of standards for linked simulators.

The MODSIM Program. The Air Force has had a modular simulator (MODSIM) design program underway for five years. The major objective of this program is to modularize the major components of flight simulators and standardize the interfaces between them. A secondary objective is to establish interfaces between simulators.

This program has just entered it's third and final phase called Proof-of-Concept. The expected end product is a military standard that is to be the foundation of future military flight simulator acquisition. Drafts of the standard should be available in 1989.

A key part of the MODSIM program is the establishment of an Interface Standards Working Group (ISWG). It is made up primarily of representatives of the flight simulator industry and government personnel involved in the project. The group has several functions. One is to monitor the work the contractor (Boeing Military Airplane Co.) and provide industry's viewpoint on the progress and direction of the project. It has no authority but can recommend changes and direction.

This group had its first organizational meeting in May 1987. It's too early to tell what impact ISWG will have, but it has excellent potential.

The *De Facto* SIMNET Standard. DARPA (Defence Advanced Research Projects Agency) has been sponsoring a program call SIMNET[3]. Prime contractors are Perceptronics and BBN. The program's goal is to demonstrate the concept of linking vehicle simulators. It has demonstrated its initial goal of linking a group of tank simulators in the same location. Future phases will add helicopter simulators (AIRNET) and fixed wing simulators (WARNET). The ultimate goal is to link large numbers of geographically separated simulated vehicles.

The program has a different aspect than what is addressed in this paper in that the program includes the simulators themselves.

The goal is to design and build relatively inexpensive simulators with just enough fidelity to train the tasks associated with multiple vehicle operation. A by-product of this approach is that the simulators are homogeneous and one does not have to deal with different levels of fidelity.

Many of the techniques developed on the SIMNET program are applicable to any linked simulator program. The SIMNET program may become a *de facto* standard for linked simulators or it may serve as the basis of a formal standard.

F-14D WST Program. The Navy, through the airframe manufacturer (Grumman), has recently awarded a contract to McDonnell Douglas to build a number of F-14D and A-6F WSTs. There is a general requirement in the F-14D WST specification that the simulators at the same site be linked[4]. It is not known at this time what the specific requirements are or how the contractor intends to approach the problem.

However the requirement is approached, this program will have a major impact of the future of linked simulators because of its size and visibility. If nothing else this program is indicative of the coming importance of multi-aircraft simulation.

Summary

Multi-aircraft simulation is a natural path for the expansion of flight simulator technology. Because of the many interests and organizations that will be involved in the development of multi-aircraft simulation, it is mandatory that the flight simulation community establish a foundation standard on which such development can proceed.

---

[1] Bosworth, L.K. (Logicon), Hecker, M-E. (Logicon), Seidensticker, S.S. (Logicon), et al, "Modular Simulator Concept Definition, Final Technical Report", USAF ASD 83-C-2201-2, July, 1984.

[2] Seidensticker, S., Johnston, D., Dolan, M., "Ethernet Performance: An Update", Unpublished.

[3] Langevin, T., Pope, A.R., Tosswill, A.R., "SIMNET Management, Command, and Control System", BBN Laboratories Inc., Technical Report 6473, March 1987.

Pope, A.R., "SIMNET Network and Protocols", BBN Laboratories Inc., Technical Report 6369, February 1987.

[4] "System Specification for F-14D/A-6F Aircrew Training Suite", Grumman Aerospace Corporation, Electronic Systems Division, A55CSACT100A, 30 June 1986.

PART-TASK SIMULATION TECHNIQUES

87-2301

Annie M. Sizer
Electronic Systems and Laboratories
General Dynamics Corporation
Fort Worth, Texas 76101

ABSTRACT

This publication will demonstrate a procedure of testing real-time software models without the use of cockpit hardware. A method to use an off-line development system to provide analysis of avionics models in an approximation of operational conditions will be presented. Utilization and techniques for an effective part-task simulation on a development system are discussed.

INTRODUCTION

Part-task simulation can be a very useful tool for the testing and integration of real-time software systems. Since a typical simulation program consists of hundreds of models and sub-models, the number of people on the team could be as many as twenty (20) or more. This requires that development and checkout strategies be devised in order that one person does not monopolize the entire simulator. Thus, this provides an opportunity for each person on the program to checkout and integrate their systems. The more part-task utilities available, the more time there will be available in the simulator for systems that are too complex to be emulated. Furthermore, an emulator can serve as a prototype for the real system to allow design development before the actual system hardware is available. Finally, it could be utilized for the development of experimental concepts.

Two typical methods of part-task simulation utilities that will be addressed are:

* real-time emulation
* off-line simulation

Real-time emulation implies conditions whereby the system is functioning as a real-time system (software characterized by communication exchanges that are controlled by the random inputs and time dependent processing of data). Off-line simulation implements the isolated modeling concepts. Thus, a single model is confined and isolated for testing, and the part-task is run off-line from the actual simulation environment.

DEVELOPMENT

Development of two part-task simulation techniques (real-time emulation and off-line simulation) will be outlined and analyzed. An important aspect in the development for an effective part-task simulation is the determination of the method (real-time emulation or off-line simulation) that would be the most beneficial for verification of the software models. After thoroughly analyzing the requirements, a plan of action to follow was developed. If the requirements are such that they require more than one model to be tested, the emulation method would be the best tool to develop. On the other hand, if only one model is required, the off-line debugging method should be considered.

REAL-TIME EMULATION

Emulation software executes as a "shell" around the test configuration, dynamically varying the test environment in response to operator or test inputs. The two models implemented were the Up-Front Controls set (UFC) and the Multifunctional Display set (MFDs) (Figure 1). This part-task simulation was developed on the Gould 32/97 processor using a Televideo 970 for a display monitor. The emulation can be operated as an integral part of the simulator when the real-time simulation is running or in a stand-alone mode. When the emulator is run with the simulation, the user-terminal inputs control the actual hardware in the real simulator.

For this part-task simulation, the software models for the UFC and MFDs have already been developed. Thus, to make this task functional, the control, display, and command software packages have to be developed. Layouts of the UFC and MFDs were constructed for display on the Televideo 970. The layout of the UFC consists of a DISPLAY section, which implements the Data Entry Display (DED), and a MENU section, which implements the Integrated Control Panel (ICP) functions (Figure 2). Since the MFD set is made up of two display units, the layout for the MFDs consists of two DISPLAY sections surrounded by MENU sections (Figure 3).

Control Software. The control software package initializes the system and simulation variables. It also activates each load module for the models and submodels to be run in the simulation. When this program finishes running, it places the system in the simulation (SIM) mode. This software program is invoked by entering the command "SIM".

Command Software. The command software executes after the program enters the SIM mode. This software package is responsible for translating command inputs (menu selection or executive commands) into formats that can be utilized by variables in shared memory (memory which is addressable from all the models). Figures 2 and 3 depict diagrams of the menu selection of inputs. The following is a list of executive commands:

* KI Kills the simulation.
* DED Displays the simulated DED and places the keyboard into the simulated ICP mode.
* MFD Displays the simulated MFD and places the keyboard into the simulated MFD mode.
* X Exit option - returns program to the SIM mode.
* Z Refresh the CRT screen (redraws the static displays - menus, rectangular outlines).
* DI Is used to display global data in shared memory.
* SE Is used to set global data in the shared memory.
* DMOD Is used to change data display mode (Octal, ASCII, Decimal, Integer, Real, Hexadecimal) for the display (DI) command.

Display Software. This software package is responsible for displaying the data from the UFC and MFD models onto the Televideo 970 screen in the models proper formats (Figures 2 and 3, respectively). The display software is made up of two parts: the static software package and the dynamic software package. The static software draws the rectangular outlines that surrounds the display of the DED and MFD formats. Also, it constructs the MENU sections for each model. The dynamic software contains a group of routines which formats the output data from the models into a form usable for display onto the screen.

OFF-LINE SIMULATION

Off-Line Simulation software executes with the isolated model (the single model isolated for debugging) acting as the nucleus for the test configuration. Only a subset of routines from the models (those routines which transmit data to or utilize data from the nucleus model) which interface with the nucleus model is executed for the part-task simulation. The UFC model was selected for the development of the Off-Line Simulation technique. This part-task simulation was developed on the Harris H1000 processor using a Wyse WY-50 terminal for a display monitor. Since this utility is run off-line from the actual simulation, the real hardware is not affected by outputs from this utility.

The layout of the UFC model was developed for display on the Wyse WY-50. The layout of the utility consists of three sections: Display, Menu, and Debug (Figure 4). The Display and Menu sections (the sections which profile the DED and ICP functions, respectively) are used to implement the UFC model. As the name implies, the debug section is used to display variables for debugging purposes. For the Off-Line Simulation technique, a software package similar to the Real-Time Emulation package was developed. Similar logic and methods in the development of Emulator software was utilized to develop the software configuration for the off-line part-task. The control, command, and display software are the three packages that make up this software configuration.

Control and Command Software. Reference the Real-Time Simulation sections for a description of the control software and command software functions. However, the executive commands for the Off-Line Simulation technique is more extensive than the commands for the Real-Time Emulation method. An outline of the executive commands will be addressed in a later section of the paper.

Display Software. This software package is responsible for displaying the static data and dynamic data for implementation of the UFC model. As I have stated before, a WY-50 was used for the display monitor. The display software consists of two software packages, the static software package and the dynamic software package. Execution of the static software draws the rectangular outline that surrounds the display of the DED, and the MENU section for the model. Execution of the dynamic software projects the UFC visible output data onto the WY-50 CRT (Figure 4).

DEBUG UTILITY

In order to analyze the manipulation of software within the models and between models, a debug software utility "Display Program" was developed. This utility will allow the display of variables containing valuable information that cannot be gathered from looking at the MFD or DED output formats. The Emulation display program is invoked by entering the DI command. It is best that this program be set up on a different terminal from the one the SIM was brought up on. However, the same terminal could be utilized for activating both the SIM and the DI modes. While the system is in the SIM or the DI mode, commands of the other program are disabled. This program can be set by invoking the "SE" command. A similar display program was developed for the Off-Line Simulation method. However, with the Off-Line display program, only ten variables can be displayed at a time.

DEMONSTRATION

A sequence of steps is taken to activate the part-task simulation tools. The control software is the initializing step in the process. This program is invoked by entering the "SIM" command. After the execution of this program, a prompt for the next command is returned.

REAL-TIME EMULATION

After entering the "SIM" command, the Real-Time Emulation program is placed in the simulation mode. All the models and submodels communicate with each other through a media called shared memory or through a local common block (a group of variables confined within the model). Reference Figure 5 for the overview of the interaction of the models. The "KI", "DED", "MFD", and "DI" commands are enterable while in the simulation mode. When the "DED" or "MFD" commands are entered, the program is placed in the simulated ICP or MFD mode respectively. The executive commands enterable while in the ICP or MFD modes are "X" and "Z". Other commands enterable are the MENU items (Figure 2 and 3). After entering the DI mode, the "SE" and "DMOD" executive commands are enterable. Reference the section under the subheading "COMMAND SOFTWARE" for a brief description of each command. On a single terminal, only one program (DED, MFD, or DI) may be displayed at a time. Thus, to make this part-task effective, either one of the above programs can be invoked on the master terminal (the terminal whereby the "SIM" command was entered) or on other terminals which are linked to the master terminal (Figure 5).

OFF-LINE SIMULATION

After the execution of the "SIM" command, the Off-Line Simulation program is placed in the simulation mode. The nucleus model communicates with the models and submodels by way of a local monitor common of shared memory. This local set up of shared memory prevents interaction with the actual simulation process. Thus, this part-task tool is run off-line from the simulation. Figure 6 presents the layout of the part-task configuration. The following executive commands are enterable after the program enters the simulation mode and the prompt symbol for the next command is displayed:

* KI — Kills the simulation.
* DED — Displays the simulated DED and places the keyboard into the simulated ICP mode.
* RE — Resets the off-line simulation.
* GO — Places the off-line simulation into the GO mode.
* ST — Places the off-line simulation into the STOP mode.
* CI — Executes a file of simulation control commands.
* GX — Repeats the last command.
* G — Places the last command upon the command line.
* DMOD — Changes data display modes for the display (DI) command. It has the following forms:

  DMOD S=sub - changes the default table used by the DI command to the specified subroutines. Cancels a previous C=com parameter.

  DMOD C=com - changes the default table used by the DI command to the specified common block. Cancels a previous S=sub parameter.

  DMOD OCT - allows all integer data items entered while in this mode to be displayed in octal. Other data items will be unaffected.

  DMOD OCT ALL - causes all data items to be displayed in octal. This affects all items.

  DMOD ASCII - allows all data items entered while in this mode to be displayed in ASCII. Other data items will be unaffected.

  DMOD ASCII ALL - causes all data items to be displayed in ASCII. This affects all items.

  DMOD NULL - cancels the previous OCTAL or ASCII display mode.

* DI — Display local data in a subroutine or global data in a common block. The item is displayed in the display window. Up to ten items can be displayed at a time.
* CL — Clears all items from the debug display window.
* SE — Sets local data in a subroutine or global data in a common block.

After entering the "DED" command, the program is placed in the simulated ICP mode, and the only executive command enterable is the "X" command, which returns the program to the simulation (SIM) mode. Other commands enterable while in ICP mode are the MENU items (Figure 4). The debug section, shown in Figure 7, illustrates the DMOD command ASCII mode. The off-line simulation utilizes only one terminal to accomplish the part-task simulation. The debug and DED formats are all displayed on the same terminal, as illustrated in the figures.

PART-TASK SIMULATION vs. COCKPIT SIMULATION

As effective as the part-task techniques are, they cannot replace the cockpit simulation. Although part-task techniques provide the capability to analyze models without utilizing the cockpit, the total effect of the simulator is not experienced. The workload of effectively interacting with several avionic models is not accomplished through the use of a part-task simulation process. From the two methods of part-task simulation addressed in this paper, the Real-Time Emulation technique will provide the most realism in comparison to the "real" simulator. The Real-Time Emulation technique has the capability to activate all of the same models activated for the running of the simulator. Thus, the primary differences as far as the system configuration between this method and the simulator are the hardware utilized to interact with the system and the monitors utilized for the display of output formats. Figures 8 and 5 illustrate for the configuration layout of the "real" simulator systems and the emulation systems, respectively. However, with the Off-Line Simulation technique, a single model is isolated

for debugging. Therefore, this method does not employ the same configuration for the avionic systems (Figure 6) as the cockpit simulator.

CONCLUSION

The part-task simulation techniques addressed in this paper have proven to be extremely useful tools in the implementation and debugging of the avionic system models. The Real-Time Emulation method and the Off-Line Simulation method were developed on two different types of computers (Gould 32/97 and Harris H1000), yet similar logic was used in the development of these two techniques. This illustrates that a part-task simulation could be accomplished on almost any type of development system. Although similar structural logic was used in the creation of the utilities, the test packages are not interchangeable. These part-task methods are of a specific nature, and the commands and displays were tailored to the needs of checkout for each system.

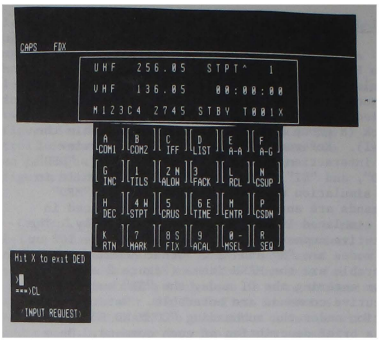


Figure 2 Part-Task Emulation (UFC)

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to A. B. Crimmins, Wallace Davis, David G. Hoelscher, Donald D. Smart, Toni L. Vaughn, and Joseph W. Warren for their valuable assistance with the organization and typing of the paper, and with the development of photographs and illustrations presented in the paper.

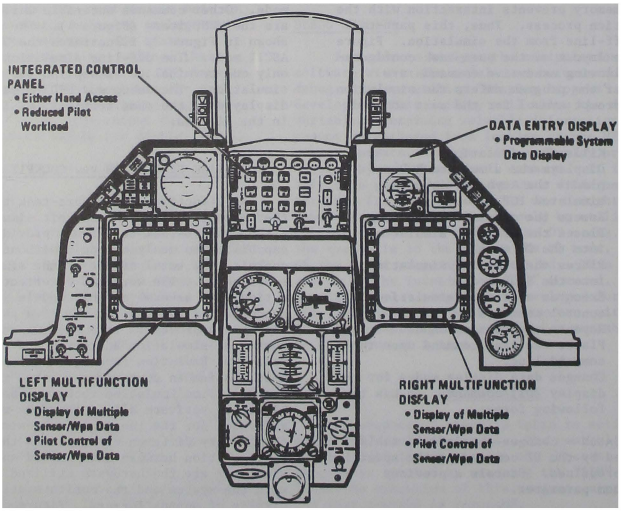


Figure 1 Cockpit Controls and Displays

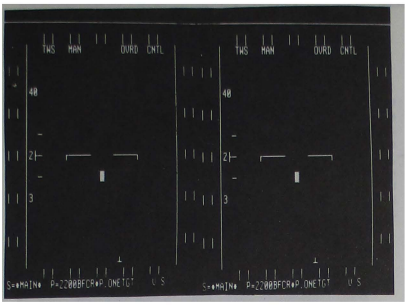


Figure 3 Part-Task Emulation (MFD)

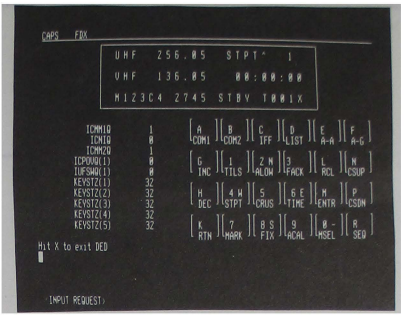


Figure 4 Off-Line Simulation (UFC)

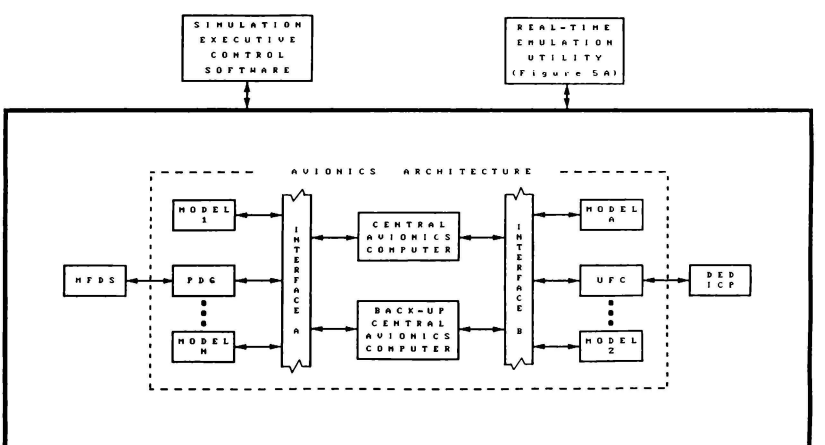


Figure 5 Real-Time Emulation Configuration

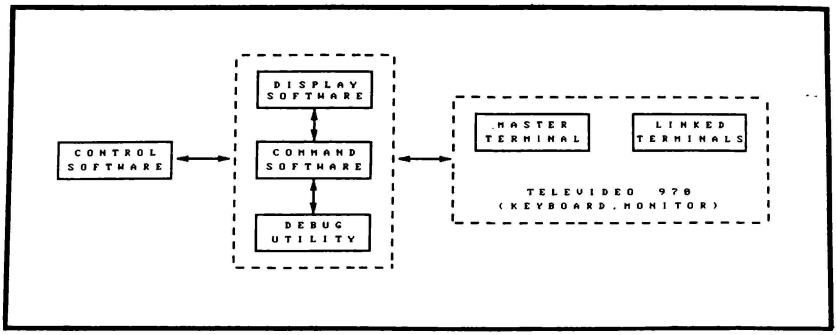


Figure 5A Emulation Software Utility

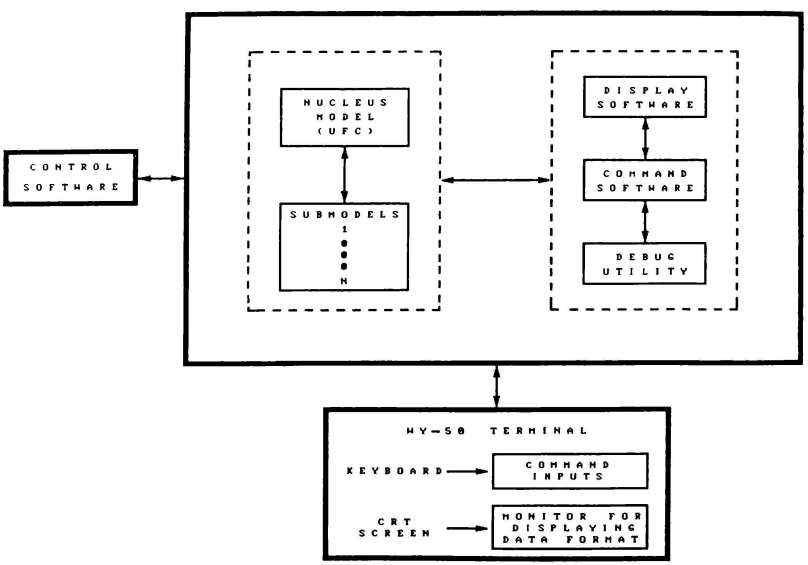


Figure 6 Off-Line Simulation Configuration

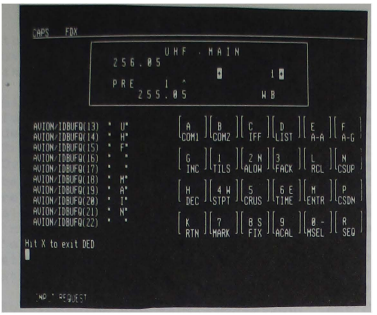


Figure 7 Off-Line Simulation (Debug Section - ASCII Mode)

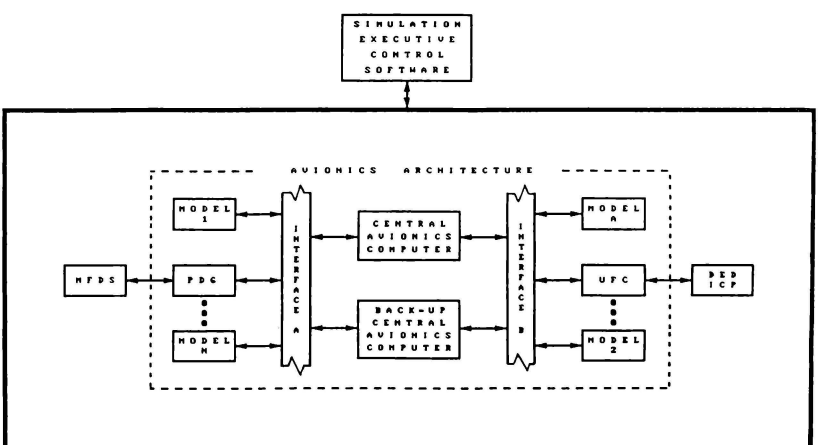


Figure 8 Simulation Configuration

APPLICATION OF PERSONAL COMPUTERS TO REAL-TIME SIMULATION SUPPORT

W. A. Ragsdale*
Unisys Corporation
Hampton, VA 23666

ABSTRACT

The advent of personal computers (PC's) provides a new cost-effective approach for analysis of simulations. This paper describes some experiences in applying this technology.

Specific examples include development of area navigation algorithms, examination of Space Shuttle abort procedures, and vehicle dynamics analysis.

Besides the obvious role of PC's as mathematical tools or terminals, it can be beneficial to simplify a large simulation to the level of a 'micro-sim'. The micro-sim can be used to project performance or examine one function of a system in detail. At the same time, the micro-sim provides a cost-effective flexible training and analysis tool.

This paper outlines some of the methods used to produce micro-sims, the results, and limitations of the method.

INTRODUCTION

In 1967 the Lunar Module Simulator (LMS), which was used to train Apollo Astronauts for landing on the moon, had a total of 576K bytes of memory, and was programmed without FORTRAN. This is roughly the equivalent of two typical desktop Personal Computers (PC's) at present.

The Microsoft Flight Simulator (Reference 1), one of the most popular programs for today's PC's, has much better graphics, sound, and data storage and retrieval systems than the LMS.

This puts a vast capability into the hands of simulation engineers, which has not been utilized to the best advantage. Part of the problem is the association of PC's with 'games'. However, the dividing line between a game and a simulation is really just a matter of opinion.

Another stumbling block has been the assumption that to be useful, a simulation must be written in a complicated language--usually FORTRAN or assembly language. BASIC, which is the defacto standard for PC languages, provides the best access to the PC's capabilities for interactive operation, graphics, sound, and input/output. Microsoft's QuickBASIC compiler (Reference 2) provides a very cost-effective engineering tool by combining interactive operation for debugging with the speed of compiled programs. The PC programmed with BASIC provides a state of the art 'slide-rule' for engineering, which should not be rejected because of its lack of programming elegance.

It is often necessary to examine one particular function of a simulation in detail, using 'driver' programs to provide typical inputs from the rest of the simulation, and to close control loops. This approach is especially useful when developing algorithms and procedures for simulations.

It is not necessary to have a detailed simulation to produce useful results. There are several possible levels of simulation. For example, pulling back on the control stick normally causes an aircraft to pitch up and climb. Various levels of simulation might respond to a pilot pulling back on the control stick in one or more of the following ways:

1. (PROCEDURES LEVEL) The pitch attitude and rate of climb on the aircraft instruments change by appropriate amounts.
2. (DYNAMICS LEVEL) The elevator moves to produce a pitching moment and other changes in aerodynamics
3. (SYSTEMS LEVEL) A detailed simulation of the hydraulic and cable systems causes the elevator to move
4. (MISSION LEVEL) The total aircraft environment, such as visual and motion cues, is simulated.

Which level of simulation is required depends on the purpose of the simulation. At the Procedures Level all that is necessary is to provide responses that look correct to the user -- with the simplest transfer functions between control inputs and observable outputs. This level of simulation is adequate for many functions (e.g. basic navigation procedures) where

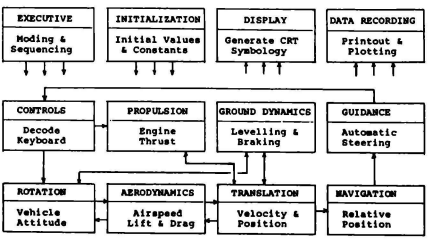


FIGURE 1. TYPICAL AIRCRAFT SIMULATION MAJOR FUNCTIONS

* Senior Member AIAA

the details of the aircraft dynamics and systems operations are of secondary importance.

If one analyzes a typical aircraft simulation, there are about a dozen major functions, as shown in Figure 1 and described in Reference 3. At the Procedures Level, and sometimes at the Dynamics Level, most of these functions can be simulated with a few equations. The result is the simplest possible aircraft simulation--for which I use the term 'micro-sim' In effect, a micro-sim is a 'simulation of a simulator', reduced to a size that can be easily manipulated and studied.

A micro-sim provides a valuable tool for training and experimentation, that can be expanded in various ways to benefit more detailed large-scale simulations. Three experiences with using micro-sims are described below.

EXAMPLE 1 -- V O R AREA NAVIGATION

Reference 4 provides the details of the VOR Area Navigation (VORNAV) research and development project at Intermetrics, Inc. in 1982. It appeared to be feasible, with existing technology, to develop a navigation system for light planes which could use the bearing data from a single VOR station, along with estimates of speed and heading, to provide area navigation capability. This would enable an aircraft to fly a straight-line path between any two desired waypoints, instead of flying from VOR to VOR. The mathematical algorithm to accomplish this is not difficult for a computer, given perfect input data. However, the bearing data has a noise level of up to 2 degrees, and the estimates of speed and heading may be incorrect.

To develop this algorithm by flying an actual aircraft was not feasible, nor did Intermetrics have access to a simulator of the type that would be needed. Instead, the algorithm was developed by expanding the navigation function of a micro-sim. The aircraft model was based on the 'K-Hawk' simulation developed by the author for a computer magazine (Reference 5) which simulated a typical light plane in less than 1 K of memory. The aircraft model was controlled with the numeric keypad, as shown on Figure 2. Detailed graphics were not required--only the basic aircraft attitude, speed, altitude, and heading information are displayed, as shown on Figure 3.

For the VORNAV project, a detailed model of VOR navigation was developed, including typical noise and geometry limitations. The VORNAV control and display panel was developed by simulating the proposed keyboard (Figure 4) with keys on the PC keyboard. The VORNAV display was simulated in the lower left corner of the screen, as shown on Figure 3.

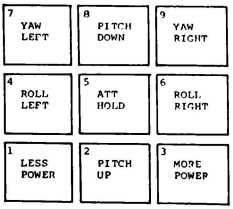


FIGURE 2. NUMERIC KEYPAD AIRCRAFT CONTROLS

K-Hawk was designed to fly in real-time in BASIC. A compiled version of the program ran about three times as fast. Nevertheless, a long cross country flight still took too long, so a 'quick-time' option was added to fly approximately six times real-time. The VORNAV program required 8 K of memory in BASIC.

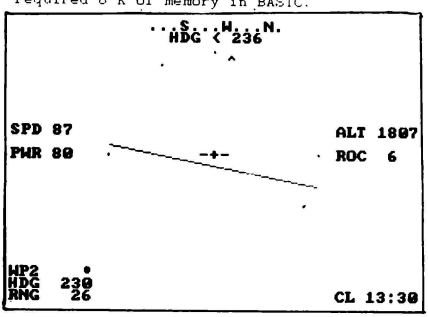


FIGURE 3. VORNAV SIMULATION CRT DISPLAY

Several algorithms and procedures were tested with the micro-sim until a solution with reasonable performance was found. The resulting algorithm was demonstrated by four flight tests in an actual aircraft, using a programmable 'Pocket Computer' and manually entering VOR bearing data. We were able to fly directly over an airport 50 miles from a VOR with the system. The proposed keyboard and display was constructed by Intermetrics for testing. U. S. Patent Number 4,577,194 (Reference 6) was granted for the VORNAV system in 1986.

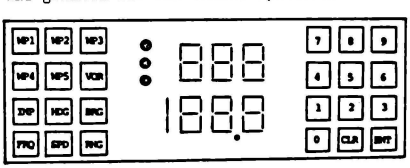


FIGURE 4. VORNAV DISPLAY & KEYBOARD

EXAMPLE 2 -- SPACE SHUTTLE ABORT IMPROVEMENTS

Reference 7 provides details of the Space Shuttle Abort Improvements Micro-Simulation (AIMS) developed at the NASA Johnson Space Center in 1985. The problem was that some existing Space Shuttle abort procedures, especially immediately after liftoff, appeared to be inadequate. There was a large matrix of possible test cases, and the full-scale simulators at Johnson Space Center were saturated with training requirements. In addition, some of the possible options could not be demonstrated without a major programming effort.

In Reference 8 the author had developed a simplified Space Shuttle dynamics model. This was expanded to produce a real-time interactive micro-sim of the Space Shuttle in the launch and abort environment. The AIMS had more detailed aerodynamics and propulsion models, as well as simplified launch and abort guidance, navigation, and control algorithms. The original specification called for the micro-sim to match a reference trajectory within 10 percent, during both a nominal launch, and a Return-to-Launch-Site (RTLS) abort. The actual results were within 3 percent.

The AIMS could be controlled either automatically, or with the keypad, or with a joystick. The display showed the pertinent flight parameters without detailed graphics, as shown in Figure 5. The AIMS also generated both numeric and graphic data so results could be analyzed and reported in detail. Figure 6 shows a typical graph generated by the AIMS.

Literally hundreds of test runs were made with the AIMS to evaluate many options for launch aborts. Those options that appeared to be most feasible were later evaluated on full scale simulations.

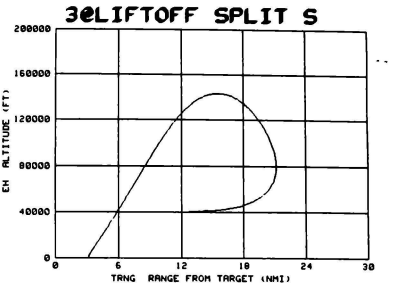


FIGURE 6. TYPICAL AIMS GRAPHICAL OUTPUT

Several needed improvements to the existing flight programs were discovered and implemented.

The most notable result was the 'Split-S' abort, following a failure of all three Space Shuttle Main Engines (SSME's) shortly after liftoff. It was demostrated that the Orbiter could climb until solid rocket booster flameout, and then perform a 180-degree pitch maneuver, without exceeding its aerodynamic limits, and return to a landing at the launch site, instead of ditching in the ocean.

It was also learned that during Vandenberg launches, the Orbiter could reach a 10,000 ft runway on San Nicolas Island in the California Channel during most of first stage.

During high inclination launches the Orbiter could reach landing sites along the East Coast, and in eastward launches, it could sometimes reach Bermuda. The envelope for these downrange aborts was explored with the AIMS.

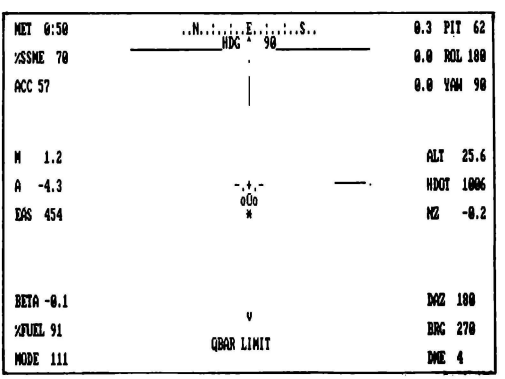


FIGURE 5. AIMS CRT DISPLAY

The BASIC program used for the AIMS was coded on both the HP-9845 and the IBM PC. The PC version was able to operate approximately in real-time after compilation. The program required a total of 21 K in BASIC.

The core of the AIMS program has also been modified to simulate a Ground Controlled Approach (GCA) for a Space Shuttle landing at a random landing site. The GCA program required 15K in BASIC.

As a feasiblility demonstration, the GCA program was modified to simulate MLS landing approaches with a Boeing 737. The 737 program required 14 K of memory.

EXAMPLE 3 -- GENERAL AVIATION AUTOPILOT INSTABILITY

The above examples represent micro-sims at the Procedures Level of simulation. The micro-sim is also useful at the Dynamics Level. An example of this occurred in 1986 during a research project at the NASA Langley Research Center.

During the development of a General Aviation Engine-Out digital flight control system, the pitch axis design exhibited instability. Since there were many variables affecting the system, and the response was not always linear, the cause could not be easily determined.

A BASIC micro-sim was developed, which provided a non-real-time simulation of the pitch axis closed loop in which the time step, integration method, control gains, and limiters could be varied. Numerical and graphic results were generated. Figure 7 shows a typical output of the program.

The problem was quickly traced to extreme sensitivity to a single gain in the control system. A variation of the pitch rate feedback gain from 2.4 to 2.5 (4 percent) made the difference between an apparently stable and unstable system. As is often the case with digital simulations, the system was stable in continuous time (or with a very small time step) but not stable in discrete time (with a time step of 1/32 sec in this instance). Having found the source of the problem, it was easy to demonstrate, using Mason's Formula (Reference 9) and Z-transform methods, that the system had an unstable root in the discrete time domain. (PC programs were also used to evaluate the roots.)

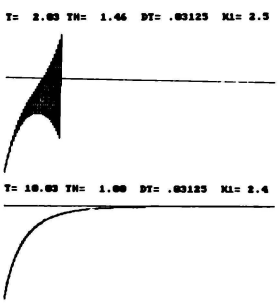


FIGURE 7. PITCH AUTOPILOT RESPONSES

The author has also developed micro-sims for a variety of other projects, including a helicopter, orbital rendezvous, and storage tank leak detection. Figure 8 is a table of micro-sims developed by the author with some of their characteristics.

MICRO-SIMULATION METHODS

The approach for developing a micro-simulation is basically the same as for any simulation, but some of the considerations are different. The following paragraphs provide an outline of steps and design philosophy for developing a micro-simulation.

1. Define the objective and groundrules. As will be described below, there are limitations to what you can do with a desktop PC. Typically, memory is not a problem, unless a lot of data is being saved. However, execution speed is at present the major limitation in developing micro-sims.

2. Define the program functions, and the inputs and outputs of each function. Use the simplest possible transfer function that relates the outputs to the inputs, based on what the user can see. Mason's Formula (Reference 9), sometimes called the 'Whammo Theorem', provides a valuable tool for simplifying transfer functions. Define constants and data requirements for implementing these functions.

3. Define the initial conditions required. How many test cases will be needed? If repeatable initial conditions (such as takeoff, over the Outer Marker, etc) are needed, define them. A menu for generating 'on-the-fly' initial conditions is essential for most projects. In addition, random initial conditions are often beneficial for training or research, so the user has to solve the problem (e.g. being lost in a navigation exercise) as he would in the real world.

4. Define the control systems or operator inputs. Interactive controls are usually essential. Many functions can be easily simulated with keyboard or pushbutton discrete inputs. Analog inputs, such as joysticks or a 'mouse' can sometimes be useful, and more typical of large scale simulations. Automated functions, especially a repeatable 'pseudo-pilot', are often beneficial.

5. Define the displays. The spectacular graphics capability of PC's is often a trap that wastes a lot of effort. All that needs to be shown is enough data to control the simulation, in the simplest possible format. Digital values are adequate for most parameters (e.g. airspeed). A few parameters, especially pitch and roll attitude, definitely require 'analog' displays. A 'visual' out-the-window scene is attractive, but rarely a requirement, and can absorb a lot of development and execution time. Written messages (and possibly aural cues) should be provided to alert the user when exceeding limits or going beyond the micro-sim's design envelope.

6. Define the operations of the simulation. How accurate must it be? It is not necessary to have accuracy beyond what is observable to the user. Error models (e.g. in sensors) should not be included unless they are a part of the problem. The update rate is often set up to match real time, but some consideration should be given to slow motion, or quick-time for long duration periods with predictable activity (e.g. on course between waypoints). Hold (freeze), Continue, and Reset (back to a specified condition) functions are nearly always needed.

7. Define data output and recording requirements. Graphic data is attractive, but expensive in terms of memory. Real time printouts of conditions at specified intervals or events are often beneficial. Data can easily be stored on diskettes for later listing or plotting. Most PC's also provide the capability of making screen 'snapshots'.

BENEFITS OF MICRO-SIMULATION

The most salient benefit of micro-sims is that they provide for interactive man-in-the-loop decisions, and thus are ideally suited for procedures development. They help to provide answers to questions such as -- what data is required to solve the problem? Is there adequate time to make those decisions? What is the best sequence of events in a given procedure?

The micro-sim also provides flexible and fast initial condition setup capability. It can be used to identify conditions or options that would require much more expense and time to discover in a large scale simulation.

BASIC allows the user to stop a program, look at any parameters, and modify the program more quickly than in any other language. This capability is most valuable during algorithm development to engineers who are not expert programmers, providing a state-of-the-art 'slide rule' , 'column pad' and 'graph paper'. Once an algorithm is developed and tested, it can be compiled or added to a large scale simulation.

The BASIC micro-sim can provide numeric, graphic, and/or pictorial results much more readily than most large-scale simulations.

Because of their low cost, PC's provide easy access to the 'hardware'. Engineers and programmers can work at PC terminals without requiring the use of expensive large scale simulations, until they have evaluated their methods and procedures.

A PC micro-sim also provides cost effective individual training. Whether learning how to fly an ILS approach, or testing the response of a pitch autopilot to a change in dynamic pressure, the micro-sim provides an educational laboratory for practice and experimentation.

LIMITATIONS OF MICRO-SIMULATIONS

There are obviously limitations to what can be accomplished with a desktop PC simulation. In his expectations, the user must be aware of these limitations. The most notable of these limitations are described below.

Controls and displays may not be representative of actual hardware. This does not necessarily lead to 'negative training' however. Realism is in the mind of the user. The important guideline is that the user must associate what he is doing with what it represents in the real world -- at least things should move in the correct direction. (A classic violation of this principle was a flight simulation 'game' in which the joystick worked backwards from the control stick in an airplane--forward was 'up'.)

The 'workload' may not be representative of the real world. For example, a pilot does not have to look at the stick to see whether he is pitching up or down. He should not have to look at the keyboard to pitch, either. This problem may be circumvented by a logical layout of the control functions on the keyboard. On the other hand, a function such as tuning a radio, requires looking away from the main instrument panel, and it is acceptable to require looking at the keyboard to tune a radio.

There are some tasks for which micro-sims are simply not adequate or applicable. To simulate the short period pitch response in real time is an example. Typically micro-sims require a time step on the order of one second. Trying to simulate the short period by brute-force integration of the classical aerodynamic equations leads to numerical instability and excessive unrealistic pitch oscillations. This means the user must either run in slow motion, or use the steady state transfer function of pitch versus stick motion. There is a tradeoff between real-time versus the integration time step versus the level of simulation. Which is most important depends on the task being simulated.

Simulations that require extensive 'table-look-ups' or iterative solutions will probably not be able to run in real-time in a micro-sim. To maintain both real-time and reasonable accuracy, micro-sims will often require numerical methods (e.g. polynomial curve fits) to simulate mathematical functions. Likewise, detailed visual scenes cannot be generated in real-time, with present technology.

CONCLUSION

The desktop personal computer can be used as a cost-effective engineering development and training tool by using simplified functional simulations -- 'micro-sims' -- to evaluate procedures and dynamics of large scale simulations.

REFERENCES

1. Artwick, Bruce, "Flight Simulator", Microsoft Corporation, 1985

2. Microsoft Corporation, "QuickBASIC Compiler for IBM Personal Computers and Compatibles", 1986

3. Ragsdale, W. A., "A Generic Desktop Aircraft Simulation", Society for Computer Simulation conference, San Diego, CA, 1985

4. Ragsdale, W. A., "VOR Area Navigation -- Techniques and Results", Institute of Navigation conference, USAFA, CO, 1982

5. Ragsdale, W. A., "Flying the K-Hawk", SoftSide Selections #44, SoftSide Publications, Amherst, NH, 1983

6. Ragsdale, W. A., "Area Navigational System", U. S. Patent Number 4,577,194, 18 Mar 1986

7. Ragsdale, W. A., "Abort Improvement Micro-Simulation Project Report", Intermetrics NV Memo 84-017, Houston, TX, 27 Sep 1984

8. Ragsdale, W. A. & Bown, R. "A Space Shuttle Orbiter Recovery Scheme Using Conventional Navigational Aids", University of Houston at Clear Lake, and 2nd AIAA Digital Avionics Conference Los Angeles, CA, 1977

9. Distephano, J.J., Stubberud, A.R., and Williams, I.J., "Theory and Problems of Feedback and Control Systems," Schaum's Outline Series, McGraw-Hill Book Co., 1967

MICRO-SIMS BY THE AUTHOR

	Simulation	Machine	Out	Language	R/T	Mem K	Use
KHAWK	Lightplane	Several	G	BASIC	Yes	1-3	Benchmark
VORNAV	Lightplane	IBM PC	B	BASIC	Quick	6	Navigation research
ACAVS	Helicopter	PDP1144	B	FORTRAN	Yes	8	Development station
AIMS	Space Shuttle	IBM PC HP9845	B B	Compiled BASIC	Yes	20	Abort procedures
GCA	Space Shuttle	IBM PC	B	Compiled	Yes	15	Guidance & navigation
B737	Boeing 737	Sperry	B	Compiled	Yes	14	MLS guid & navigation
KAL2D	Kalman Filter	IBM PC	G	Compiled	No	8	Training & demo
WFRDL	Orbital Rendezvous	IBM PC	G	Compiled	Quick	6	Training & demo
SRVSTN	Service Station	Compaq	G	Compiled	Quick	12	Algorithm research
FLONET	Flow Network	IBM PC	N	BASIC	No	8	Algorithm research
GENESYS	Generic System	TRS80	B	BASIC	No	5	Training & demo
GAAP	Lightplane Autopilot	Sperry	G	BASIC	No	4	Dynamics analysis

KHAWK has been modelled on the TRS-80, Tandy Color Computer, Tandy 1000, TI-99, Commodore 64, Sinclair/Timex ZX80, HP9845, IBM, TI, Compaq, AT&T, Seequa, and Sperry PC's

OUTPUT G=graphic N=numeric B=both

FIGURE 8.

87-2370

EFFECT OF TIME DELAY ON MANUAL FLIGHT CONTROL AND FLYING QUALITIES DURING IN-FLIGHT AND GROUND-BASED SIMULATION

Randall E. Bailey* and Louis H. Knotts**
Calspan Advanced Technology Center
Buffalo, New York 14225
and
Capt. Scott J. Horowitz† and Capt. Hugh L. (Pat) Malone III††
Air Force Human Resources Laboratory
Williams AFB, AZ 85224

INTRODUCTION

During the flight testing of the F-16, F-18, Tornado, and Space Shuttle vehicles, potentially disastrous pilot-induced oscillations (PIO) were encountered (Reference 1). These severe flying qualities deficiencies were largely attributed to the delay between the cockpit control input and aircraft response introduced by the flight control systems. These incidents spurred considerable research investigating the effects of time delay on flying qualities.

From this research, 100 msec has been established in the military specification for piloted vehicle flying qualities (MIL-F-8785C) as the allowable maximum delay between cockpit control input and aircraft response for satisfactory Level 1 handling qualities. Level 2 and Level 3 upper limits were established at 200 and 250 msec delay, respectively. These requirements are independent of aircraft size (classification) and aircraft mission, although this universality may be conservative. Recent studies have shown that for large aircraft (Reference 2) and for relatively benign tasks (Reference 3), larger delays than those allowable in MIL-F-8785C can be tolerated before a degradation in flying qualities occurs.

Time delay is not only a problem in flight control design, but also in ground-based simulation design. In simulators, for example, delays can occur between the cockpit control input and visual/motion response due to finite digital computational times (pure delays), digital integration techniques, pipeline architectures, and anti-aliasing filters. The latter delay contribution is an example of an "equivalent" time delay where the phase lags introduced by these filters are essentially identical to a pure delay element in the frequency range of interest to the pilot. Equivalent time delay is a convenient method by which the delay of a system can be derived and hence, specified or measured against criteria. The definition and measurement of time delay are important technical issues; however, they will not be treated in this paper.

The effects of time delay, introduced during ground-based simulation, should be similar in many respects to the in-flight effects. Ground simulator investigations have been reported in several studies, although typically these studies were deficient in generality. The effects of time delay in ground simulation are not independent of simulation device particulars such as visual scene content, field-of-view, motion system design, and motion-visual cue synchronization, for instance. Allowable time delay limits have evolved informally in simulator design specifications based on these specific studies, past experience, and also from the flying qualities military specification limit of 100 msec (Reference 4).

Despite these efforts and specifications, time delays in ground simulation are still cited as a problem during piloted evaluations. Time delays clearly affect the transfer of training process adversely and lead to a degradation in simulated handling qualities with erroneous results. The problem clearly becomes that of defining accurately the amount of delay that can be tolerated in a simulation without affecting flying qualities, manual control behavior, and transfer of training.

This paper describes the results of an experiment whose goals were the generation of guidelines and development of a data foundation for the specification of allowable time delay in ground-based simulators. The effects of time delay on flying qualities and manual flight control were investigated during in-flight simulation where "perfect" motion cuing is available. Because the in-flight simulator also had the capability to serve as a ground simulator cab, a ground-based, no-motion replication of the in-flight experiment was also performed. The experiment will permit investigation of simulator motion requirements by examining the extreme conditions of motion versus no-motion. The visual cues for both the ground and flight phases of this study were limited to a head-up display. This contrasts with previous in-flight investigations that used full field-of-view, "perfect" visual environments. Subsequent investigations can build from this foundation regarding the influences of ground simulation device particulars (e.g., visual scene fidelity, motion base washouts, etc).

EXPERIMENT

The specific objectives of this experiment were to:

- Investigate the effects of time delay on flying qualities and manual control during flight.
- Replicate this experiment using a no-motion ground based facility as an initial extension of the in-flight data oriented toward ground-based simulation device particulars.
- Provide the requisite data foundation for associated transfer of training studies and manual flight control analyses.

* Senior Engineer, Member AIAA
** Principal Engineer, Member AIAA
† Chief Engineering Evaluation Pilot, Member AIAA
†† Chief Engineering Evaluation Pilot, Member AIAA

Experiment Variables

From the results of previous in-flight studies, the effects of time delay have been reported as a function of the vehicle mission for specific piloting task demands and control response characteristics (e.g. short period frequency and control authority). Given these parameters, a vast experiment matrix was possible. As a first experiment in this technical area, a broad impact, "shotgun" experimental approach was taken rather than a parametric variation approach. Accordingly, the effects of time delay were investigated using four aircraft configurations which, in some broad manner, characterized the gamut of military ground-based simulator applications. The configurations and their attendant missions were representative of such aircraft as:

"F-16" - small (fighter) aircraft; demanding task environment

"C-21" - small (transport) aircraft; benign task environment

"C-17" - large (transport) aircraft; demanding task environment

"C-141"- large (transport) aircraft; benign task environment

The experiment provided a representative simulation of the handling characteristics of these vehicles in either demanding or benign piloting task environments as required. Only a representative simulation of each class of vehicle was intended using data available in the public domain to develop the generic simulations. The data were used to derive typical feel system and aircraft response dynamics characteristics (e.g., equivalent short period frequencies and dampings). Inherent delays that may be present in the simulated vehicle were not included in the simulation. The configuration names are retained only because they are expedient descriptions of the simulated vehicle type, its handling characteristics and piloting task demands. The flying qualities results, presented later, are not necessarily representative of the actual vehicles. In this experiment, a centerstick controller was used exclusively even though the actual C-21 and C-141 aircraft have wheel controllers and the actual F-16 aircraft are equipped with sidesticks.

Pertinent configuration characteristics are presented in Table 1. Since this experiment was an investigation of degrading time delay effects, the simulated vehicle handling characteristics, when flown during their attendent evaluation tasks, were designed to produce Level 1 flying qualities with no added time delay. The lateral-directional characteristics of the configurations were tailored to be good and unobtrusive; thus, "feet-on-the-floor" roll maneuvers could be performed. This allowed the pilot to concentrate on pitch and roll control without objectionable yaw or sideslip.

Using this experimental approach, three variables were tested:

- Time Delay
- Aircraft Configuration (control response characteristics and mission)
- Motion Cuing

The schematic diagram of this set-up is shown in Figure 1. The experiment matrix consisted of five values of pure time delay (added to the simulation) ranging from 0 to 240 msec. Delay was, of course, the primary experiment variable, and it was introduced in both the pitch and roll axes. The maximum delay values were selected to provide, at worst, Level 3 handling qualities, but not so poor that aircraft control would often be in question.

Table I
Summary of Configuration Characteristics

AIRCRAFT	ζ_{sp}	ω_{sp} (rad/sec)	(F_{es}/g) (lbs/g)	(lbs/in)	τ_R (sec)	(F_{as}/δ_{as}) (lbs/in)
F-16	.70	6.30	10.0	8.0 B.O.= 1 lb	.35	3.5 B.O.= .5 lbs
C-17	.70	2.70	16.0	11.0 B.O.= 1 lb	.65	3.75 B.O.= .5 lbs
C-21	.40	4.0	30.0	12.0 B.O.= 2 lbs	.87	6.0 B.O.=.75 lbs
C-141	.40	2.0	30.0	12.0 B.O.= 2 lbs	1.0	6.0 B.O.=.75 lbs

B.O. = Breakout Force

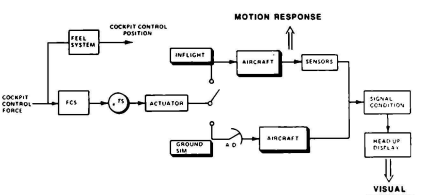


Figure 1 Experiment Set-Up

Test Vehicle

The test vehicle was the USAF/Flight Dynamics Laboratory NT-33A variable stability aircraft. The NT-33 aircraft was modified and is operated by the Calspan Corporation under USAF contract as an in-flight simulator. The front seat control system of the NT-33A has been replaced by a full authority fly-by-wire flight control system and a variable response artificial feel system. The evaluation pilot, who sits in the front cockpit, controls the aircraft through a standard centerstick and rudder pedal arrangement or a sidestick controller. A fully programmable head-up display (HUD) is installed in the front cockpit.

The front seat, fly-by-wire control system and variable response feel system were programmed as required by the experiment objectives. The system operator in the rear cockpit, who also acts as safety pilot, controls the simulated HUD and aircraft configuration characteristics. During this experiment, the evaluation pilot had no prior knowledge of the configuration characteristics.

In this investigation, the head-up display was programmed to generate the piloting evaluation tasks. Instrument meteorological conditions (IMC) were simulated using a blue/amber vision restriction system to

limit the visual cues available to the pilot. The evaluation tasks were thus repeatable, known quantities similar to those used in previous manual control laboratory experiments. The instantaneous field-of-view (FOV) of the HUD and hence the visual FOV available to the pilot was approximately 20°.

The NT-33 aircraft is normally operated as a 3 degree-of-freedom in-flight simulator utilizing response feedback. With this methodology, the NT-33 vehicle stability and control characteristics are augmented by the appropriate feedforward and feedback variable stability system (VSS) signals and gains to achieve the desired simulation dynamics by appropriate deflection of the NT-33 elevator, aileron, and rudder control surfaces. It is important to note that the evaluation pilot cannot feel the control surface motions of the NT-33 necessary to achieve the desired simulation. Cockpit control feel is provided by an electrohydraulic feel system.

A fixed-based ground simulation capability is provided by appropriate interfaces to the NT-33 aircraft. The ground simulation utilizes the actual aircraft hardware with the exception that the NT-33 motion responses and sensors are replaced by a real-time computer simulation, interfaced into the fly-by-wire flight control system (Figure 2):

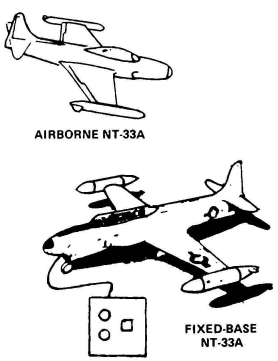


Figure 2 In-Flight and Ground Simulation Using NT-33

- External electrical and hydraulic power are provided. This provides for normal operation of the NT-33 systems and allows use of all hardware elements of the NT-33.

- The HUD, under simulated IMC, provides the visual cues to the pilot.

- The NT-33 control surface motions are provided as inputs to a PDP-1144 digital computer. Based on these inputs, the NT-33 equations of motion are calculated in real-time and the computed motion states are output to the vehicle where they are inserted in the variable stability system in place of the aircraft sensors. Thus, the NT-33 variable stability system is operated the same as it is in-flight, with the exception of aircraft motion.

To the evaluation pilot, the primary difference between the in-flight and ground simulation is the absence of motion cuing. (Certainly, other factors are also not present, such as aural cues and stress.) The in-flight and ground-based simulations were mechanized such that the equivalent time delay between the cockpit control input and the HUD (visual) attitude response was identical during in-flight or ground-based simulation. Without any experimentally added delay, this was a constant 100 msec. During in-flight simulation, the visual (HUD) response lagged the motion response by 45 msec equivalent delay.

Evaluation Tasks

Three types of HUD-displayed evaluation tasks were used for this experiment. The format of the HUD is presented in Figure 3.

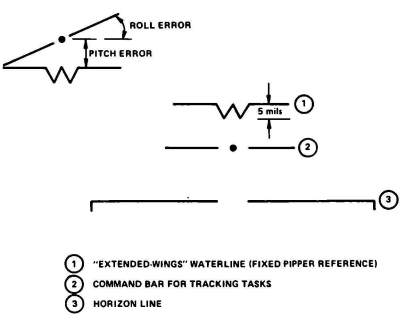


Figure 3 Head-Up Display (HUD) Format

The tasks evolved from a marriage of flying qualities and manual control concerns; that is, the evaluation tasks were specifically selected to provide a flying qualities "test" as well as generate data for subsequent pilot modeling and manual control analysis. The evaluation tasks included:

- Step-and-ramp compensatory tracking:

  This task represented a good flying qualities test maneuver and was flown on each evaluation. The command bar moved in a series of step-and-ramp attitude commands in both pitch and roll to lead the pilot through a flight profile. The "discrete" attitude commands provided the pilot a clearly discernible yet demanding task command from which to judge the pilot-vehicle performance and hence, flying qualities.

- Sum-of-sines compensatory tracking:

  Compensatory attitude tracking tasks were performed against a sum-of-sines generated command. Single axis (pitch-only and

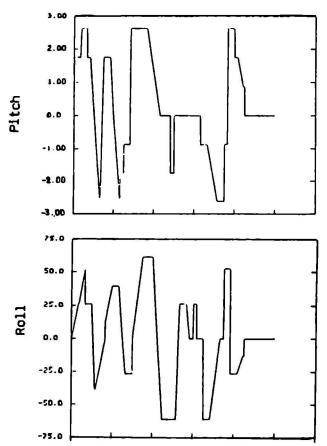
roll-only), and combined axis tracking were used. These tasks were primarily intended for subsequent manual control analysis.

- Sum-of-sines disturbance regulation:

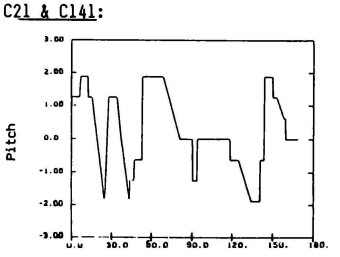
  Sum-of-sines commands inserted directly into the pitch and roll control surface actuators were used for the disturbance regulation task. This task was intended to contrast the attitude tracking tasks and to highlight motion cuing effects.

The sum-of-sines task attributes were defined in a pre-experiment analysis. All three tasks were tailored to the intended missions for each vehicle type. For instance, the demanding F-16 and benign C-141 and C-21 step-and-ramp tasks are shown in Figure 4. The F-16 task required high-g maneuvers and aggressive pitch and bank angle captures; conversely, the C-141 and C-21 task was limited to less than 45° bank angle commands and relatively small pitch attitude commands. The F-16 and C-141 sum-of-sines tasks are presented in Figure 5 and show similar differences.

F16:



C21 & C141:



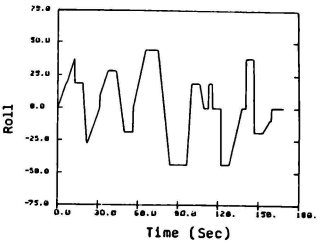
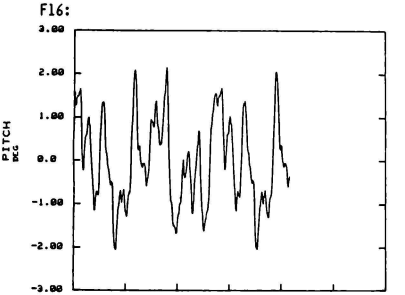


Figure 4 Discrete Tracking Task Example

F16:



C141:

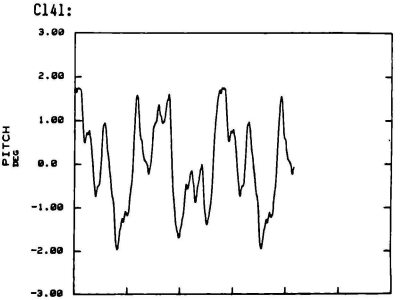


Figure 5 Sum-of-Sines Example

The task difficulties were tailored to each vehicle and its attendent mission during calibration flying. This determination was not made based on the no-motion ground simulations. The objective was to ensure that:

- Each configuration with no added delay exhibited Level 1 flying qualities in its attendant evaluation task scenario/mission during in-flight simulation.

- The evaluation tasks roughly portray the intended task difficulty of the mission for each vehicle type.

RESULTS

Three experimental test pilots acted as evaluation subjects. Each was a trained flying qualities evaluation pilot. Before data were collected, each pilot received four flights in the NT-33 ground-based simulator for training.

An evaluation flight consisted of five evaluations comprised of one aircraft configuration with five values of added time delay. The experimental delays, ranging from 0 to 240 msec, were randomly ordered. The identical experiment protocol was used for both in-flight and ground-simulator evaluations.

Eight flights and eight ground simulations for data were planned for each pilot; thus, each pilot would replicate every evaluation in both simulation phases.

The experiment data consisted of pilot ratings, using the Cooper-Harper pilot rating scale (Reference 5), pilot comments, and task performance records. The task performance records included aircraft and pilot control parameters recorded on an onboard digital recorder, voice recordings, and video tape. The pilot comments were made with reference to a formal pilot comment card. Pilot ratings should not be viewed without full regard to the attendant pilot comments.

Effects of Time Delay: In-Flight Simulation

Using the Cooper-Harper pilot rating scale (Figure 6), Level 1 flying qualities are defined as being satisfactory without improvement with pilot compensation not a factor (PR < 3.5); conversely, Level 2 flying qualities exhibit deficiencies which warrant improvement. Pilot compensation is at least moderate and pilot-vehicle performance is adequate or desired at best (3.5 < PR < 6.5). Desired and adequate performance standards were clearly defined and these standards were identical for both in-flight and ground-based simulation.

The maximum tolerable delay introduced in a simulation can be defined as the maximum delay before which flying qualities degrade to Level 2 (given that Level 1 flying qualities existed without added delay). Implicit in this rating difference is a change in pilot control strategy, behavior, and/or workload. In this paper, flying qualities results are discussed primarily. The specific effects of time delay on manual flight control will be presented briefly, although a more detailed treatment is left for a companion paper (Reference 6). Specific transfer of training issues were not addressed in this experiment, albeit this topic is implicitly included in the flying qualities results.

In Figure 7, the pilot rating results are presented for each aircraft as a function of the added experimental time delay. The overall delay, from cockpit control input to (HUD) visual response equals the experimental added time delay plus 100 msec.

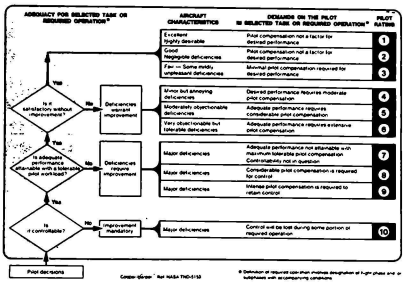
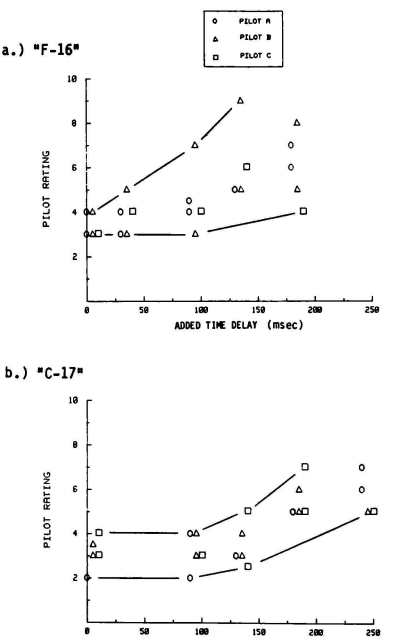
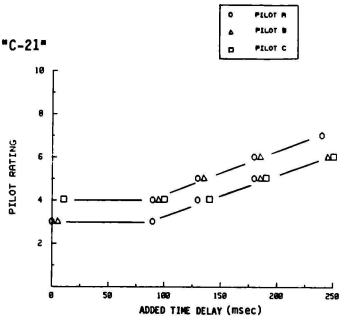


Figure 6 Cooper-Harper Pilot Rating Scale

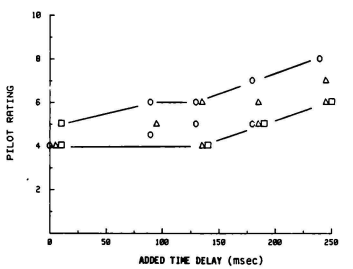


Figures 7a, 7b Effect of Time Delay on Flying Qualities for In-Flight Evaluations

c.) "C-21"



d.) "C-141"



Figures 7c, 7d Effect of Time Delay on Flying Qualities for In-Flight Evaluations

The pilot ratings are for the overall tasks including the "discrete" step-and-ramp, command sum-of-sines, and disturbance regulation tasks. In general, the pilots found that, for the discrete task, they could only attain satisfactory performance using normal closed loop flying techniques. The discrete task encompassed both gross and fine tracking. The combined axis command and disturbance regulation tasks were also good tasks in that they also required normal closed-loop piloting techniques. These tasks were only fine tracking exercises, however, which were insufficient in exposing gross acquisition flying qualities. The pilots often adopted specialized techniques to achieve good tracking performance during the single axis sum-of-sines command task. These tasks consisted solely of small perturbations in a single axis which allowed the pilot to concentrate fully on the task.

The pilot rating results for the no added delay cases, indicate that Level 1 ratings were achieved on the average for the F-16, C-21, and C-17 configuration. Despite the very benign task, Level 2 flying qualities on the average (PR ~ 4.0) were the norm for the C-141 configuration. Level 2 ratings were given for the C-141 because the response was sluggish and the stick forces were high.

The pilot rating trends with added time delay show a general degradation. In the case of the F-16, the rating trend is distorted by the learning effects of one pilot (Pilot B). On his first evaluation flight (after four training ground simulations), Pilot B gave worse ratings at each added time delay point than the other two pilots or himself on a subsequent flight. These ratings were caused by learning effects and may be indicative of potential problems associated with training simulations for a highly responsive vehicle. During in-flight simulation, the F-16 configuration required the pilots to be "low gain" or "light-on-the-stick" because of the low stick force requirements and quick initial pitch response. It was likely that Pilot B carried over inappropriate control behavior from the initial training ground simulations to the flight environment. In the ground simulator, the abrupt response and the low stick force requirements of the F-16 are not apparent due to the absence of angular and normal acceleration cues. On his first flight, poor performance and a severe flying qualities degradation resulted because of his objection to and/or his inability to adapt to the special control behavior required of this configuration.

Discounting this one flight, the remainder of the pilot ratings were almost unanimously within the generally accepted range of ±1 rating unit. This consistency is particularly good for the C-21 configuration. For the two transport configurations flown in a benign task environment (C-21 and C-141), pilot ratings appear to degrade at a constant rate with added delay (approximately 1.5 rating units per 100 msec added delay). A threshold exists in the data where, below a threshold value of added delay, flying qualities (pilot ratings) do not degrade. This threshold is approximately 130 msec. For the two aggressively flown aircraft (F-16 and C-17), the rate of pilot rating degradation with time delay appears to be slightly higher than the transport vehicles, although not significantly so.

For each aircraft, as the time delay became significant, control problems became evident with increasing tendencies toward overshoots, oscillations, and PIO. These deficiencies are evident in the pilot ratings, comments, and task performance records. To achieve any degree of pilot-vehicle performance with additional delay, each pilot adopted his own particular compensation techniques. The techniques varied somewhat for each configuration; however, the techniques were uniform in their attempts to achieve adequate response time, yet avoid PIO and attain adequate damping (minimize bobbling and overshoots). For instance:

- Pilot A avoided large control inputs and response rates ("lagged behind target", "cut aggressiveness") until the target stopped whereby he used lead compensation or went "open loop" momentarily to settle on the target.
- Pilot B used high frequency, pulsing-type inputs ("dithering") whenever possible. Other techniques were to back out-of-the-loop and lower his control gain.
- Pilot C primarily used control input shaping or "tightened his grip" on the stick which had the effect of imposing small, smoother control inputs than what he may do normally.

Although the pilots' control techniques vary significantly, their goal was the same and their performance in terms of Cooper-Harper ratings was remarkably consistent. In terms of simulator training, if a simulator contains significant delays, these specialized techniques will be learned by the trainee and erroneously taken to the flight environment.

The pilot performance during the step-and-ramp task with the F-16 is shown in Figure 8, for the 0, 90, and 180 msec added delay cases. Time-on-target (TOT) and normalized rms (root-mean squared) error (NRMSE) scores are plotted. The normalized rms error score is the rms pitch error divided by the rms pitch angle command. The time-on-target is the cumulative time that the error was less than 5 mils. The correlation of TOT and rms error has been shown previously to roughly distinguish flying qualities (Reference 7). The data from this program indicates a hyperbolic trend where the product of NRMSE and TOT is constant. Pilot A showed the most repeatability for his two F16 flights. Pilot B, as noted earlier, demonstrated significant learning effects where his first flight was significantly worse (lower TOT and higher NRMSE). Despite these differences, Pilot A and Pilot B both demonstrated consistency with configuration changes in that, as the added time delay increased on a flight, the TOT decreased and the NRMSE increased. Pilot C demonstrated great adaptability in control technique by attaining almost identical performance when added delay increased from zero to 180 msec.

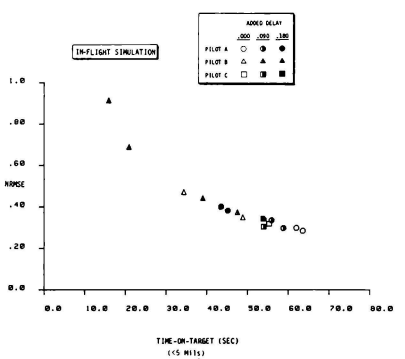


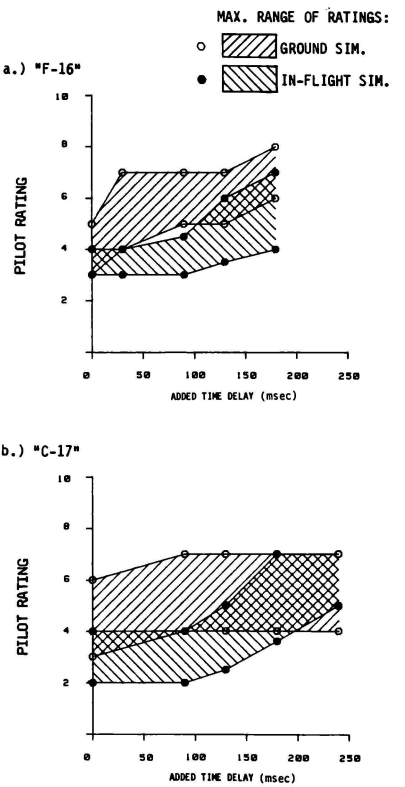
Figure 8 Time-On-Target Versus Normalized RMS Error Data for In-Flight Simulation

Effects of Time Delay: Ground Simulation

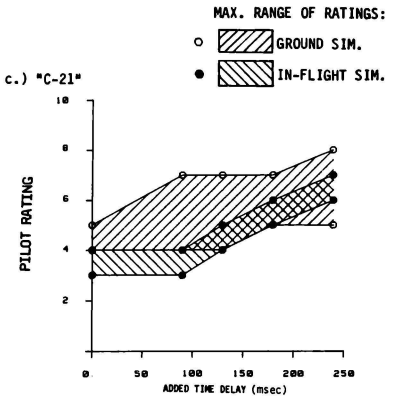
A no-motion ground-simulation was provided as a replication of the in-flight experiment. Evaluations were flown using the blue/amber vision system to restrict the visual cuing available to that of the head-up display. This small foveal visual field is identical to the in-flight, full-motion phase.

The optimum experimental procedure would have selectively interwoven the flight and ground simulator data flights. Unfortunately, logistics prevented this experiment design and the flight and ground simulations were interspersed by availability and practicality. The effects of the irregular interspersion are not known.

In Figure 9, the range of pilot ratings for the ground simulator are shown compared to those obtained during the flight phase of the experiment. The degradation of flying qualities with added experimental time delay was not nearly as dramatic as with the in-flight evaluations. A slight degradation was noted but a "threshold", where flying qualities are unchanged with added time delay, was not apparent. The degradation was generally at a lower rate (1.0 PR unit per 100 msec) than the in-flight evaluations.



Figures 9a, 9b Comparison of Ground-Based and In-Flight Simulation Pilot Ratings



Figures 9c, 9d Comparison of Ground-Based and In-Flight Simulation Pilot Ratings

For no added delay, none of the configurations were evaluated, on the average, as being satisfactory without improvement. In fact, only three ratings of Level 1 were given in the entire ground-based simulation phase. Desired performance could be achieved on occasion during ground simulation, however, the pilot workload was greater than minimal and Level 2 (PR ~ 4.0) flying qualities resulted.

The most difficult piloting task during ground-based simulation was clearly gross acquisition during the discrete tracking task for the aggressively flown aircraft (the F-16 configuration, in particular). This difference between ground and flight evaluations for the F-16 is very clear on Figure 9a. It was often remarked that gross maneuvering with the F-16 was a "guess". The pilots, without any motion cuing, did not know how hard to pull to close on the target. The combination of an abrupt pitch response, light stick forces, and the lack of 'g' cuing probably contributed to this deficiency. As one might expect, the addition of time delay accentuated the gross acquisition difficulties.

The predominate pilot comments for large added delays were directed at significant overshoots and PIO. The same degree of control difficulties were not encountered for the C-21 and C-141 discrete maneuvering tasks, as for the aggressively flown aircraft. Of course, the stick forces are significantly heavier for these vehicles and the task demands are also significantly less. Not surprisingly, the pilot rating differences between flight and ground are greatest for the F-16 and least for the C-141.

In Figure 10, the task performance for the F-16 in the ground simulation, step-and-ramp task is presented. These data are significantly different than the in-flight data presented in Figure 8. The clear tradeoff between TOT and NRMSE that is shown in Figure 8 is not apparent in Figure 10. Also, the consistency of decreased TOT and increased NRMSE with time delay is not evident in Figure 10. These data would suggest that without the in-flight cues, the pilots are more inconsistent in task performance and/or control behavior.

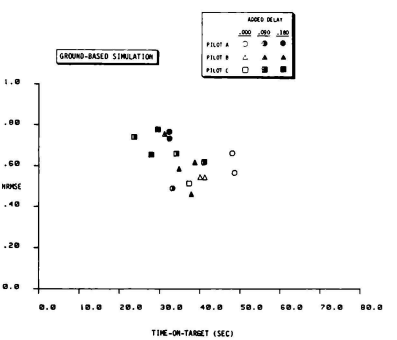


Figure 10 Time-On-Target Versus Normalized RMS Error Data for Ground-Based Simulation

CONCLUDING REMARKS

A specification for maximum allowable simulator time delay must accurately reflect the flying qualities effects as well as the attendent issues of transfer of training and quantified pilot control behavior variations. The results of an in-flight experiment investigating the effect of time delay on flying qualities have been presented. Cooper-Harper pilot ratings also reflect, at least qualitatively, the control behavior variations caused by time delay. Given the results of this experiment, total delays of up to 150 msec (100 msec plus 50 msec experimental added delay) are tolerable in a simulation environment. This delay requirement reflects the most stringent condition of a highly maneuverable, highly responsive fighter

aircraft simulation flown in a demanding task environment. The 150 msec requirement may be conservative and hence, relaxed for less responsive aircraft or vehicles flown in less demanding tasks (such as the C-21 or C-141 configurations shown here). For this specification requirement, the total delay reflects the "equivalent" time delay from cockpit control input to visual system response. It should be noted that this delay specification arises from a complete, full fidelity non-attentuated motion simulation.

The in-flight experiment was replicated using the NT-33A as a ground simulator to investigate the effects of time delay when no motion cues are available. Significant flying qualities differences were shown to exist particularly for a highly responsive, aggressively flown aircraft. The effects of motion are least significant for a sluggish aircraft flown in a benign mission/task environment.

ACKNOWLEDGEMENTS

This paper was drawn from work performed under Air Force Contract No. F33615-83-C-3603. Mr. Steve Markman was the Air Force contract monitor.

REFERENCES

1) Smith, R. E. and Bailey, R. E., "Effect of Control System Delays on Fighter Flying Qualities", Paper presented at AGARD Flight Mechanics Panel Symposium on "Criteria for Handling Qualities of Military Aircraft", Fort Worth, Texas, April 1982.

2) Meyer, R. T.; Knox, J. R.; and Tingas, S.A.; "Suggested Revisions to MIL-F-8785C for Large (Class III) Aircraft", AFWAL-TR-83-3015, February 1983.

3) Berry, D.T. et al, "A Summary of an In-Flight Evaluation of Control System Pure Time Delays During Landing Using the F-8 DFBW Airplane", AIAA Paper No. 80-1626, August 1980.

4) Woycechowsky, B.J., "Flight Simulation Cue Synchronization", Paper presented at 1983 IEEE National Aerospace and Electronics Conference, Dayton, Ohio, 17-19 May 1983.

5) Cooper, G.E. and Harper, R. P. Jr.; "The Use of Pilot Rating Scale in the Evaluation of Aircraft Handling Qualities", NASA-TND-5153, April 1969.

6) Levison, W.H. and Papazian, B., "The Effects of Time Delay and Simulator Mode on Closed-Loop Pilot/Vehicle Performance: Model Analysis and Manned Simulation Results", Paper presented at AIAA Flight Simulation Techniques Conference, AIAA Paper No. 87-2371CP, Monterey, California, August 1987.

7) Onstott, E. D. and Faulkner, W.H., "Prediction, Evaluation, and Specification of Closed Loop and Multiaxis Flying Qualities", AFFDL-TR-78-3, February 1978.

THE EFFECTS OF TIME DELAY AND SIMULATOR MODE ON CLOSED-LOOP PILOT/VEHICLE PERFORMANCE: MODEL ANALYSIS AND MANNED SIMULATION RESULTS

William H. Levison*
Bruce Papazian

BBN Laboratories Inc.
Cambridge, Massachusetts

Abstract

The optimal control model for pilot/vehicle systems was used in the design of a manned simulation study performed by Arvin/Calspan and in the interpretation experimental results. Experimental variables included (a) control system delay, (b) simulated aircraft dynamics, (c) simulator mode (ground base or in-flight). Rms error trends observed experimentally generally conformed to pre- and post-experiment model predictions. The (adverse) effects of delay on tracking performance were slightly greater for a simulated high-performance fighter flown agressively than for a simulated heavy transport flown in a less demanding task; delay effects were somewhat greater in the ground simulator than in flight; and differences between in-flight and ground-simulator performance were relatively small for tasks with no added delay. There was some evidence of pilot response nonlinearity. The generally good agreement between predicted and experimental performance metrics (both rms errors and pilot frequency response) suggests that a viable technique for determining time delay requirements can be based on the joint use of simulation and model analysis.

Introduction

This paper summarizes the results of the analysis performed by BBN Laboratories Inc. (BBN) in support of a study by Arvin/Calspan to explore the effects of control-system delay on pilot opinion ratings and manual control performance in an attitude tracking task. In addition to time delay, the principal experimental variables of the Calspan study were simulated aircraft type (ranging from fighter aggressively flown to heavy transport nonagrressively flown), simulator mode (ground-base or in-flight), type of external forcing function (command or disturbance), and point of application of the external input (roll-axis only, pitch-axis only, or combined pitch and roll). Details of the Calspan experimental study are provided in a companion paper[1]; a more comprehensive documentation of the experimental study and model analysis are provided by Bailey et al.[2]

The optimal control model (OCM) for pilot/vehicle systems was used to (1) aid in the design of the Calspan experiments,(2) help interpret experimental results, and (3) explore the utility of the OCM as a tool for interpolating the results of simulation experiments designed to explore time delay effects. The reader is assumed to be familiar with the structure and application of this model[3-6].

Pre-experiment model analysis was performed to select parameters for the external forcing functions and to predict the effects of time delay on rms pitch and roll tracking error as a function of aircraft dynamics, input type, and simulator mode. This analysis predicted:

1. modest increases in tracking error score with increasing delay,
2. larger delay effects for the generic high-performance fighter flown aggressively ("F-16") than for the generic heavy transport ("C-141") flown in a more relaxed manner,
3. relatively small differences (less than 10%) between performance in-flight and in the ground-based simulations, and
4. similar relative effects of experimental variable on performance in the pitch and roll axes.

As shown in this paper, these pre-experiment predictions were largely confirmed by the manned simulation experiments.

Post-experiment analysis was limited to a subset of the Calspan experimental conditions. Pitch- and roll-axis performance in single-axis target-tracking tasks was analyzed.

"Single-axis" tasks were tasks in which an external forcing function was applied to either the roll or the pitch axis, but not both. Forcing functions were applied to both axes in the "combined-axis" tasks. In all cases, the pilot was required to control both the pitch and roll axes.

Analysis Procedures

Statistical Analysis of Experimental Data

Means and standard errors were obtained for closed-loop performance metrics and for pilot frequency response measures. Within-trial mean and standard-deviation (SD) scores were first computed for all important problem variables. These scores were then averaged across the two replications/subject to provide average performance measures for each subject for each

*Member AIAA

of the experimental conditions considered in the BBN analysis. Finally, population means and estimated standard errors of the population means were computed for each condition of interest. The summary statistics reported in this paper are the means and standard errors of the SD scores for tracking error and control force.

Pilot frequency-response metrics consisted of the describing function -- shown as "gain" (more properly, "amplitude-ratio") and "phase-shift" curves, -- and curves of the pilot's "remnant" (the spectral density of that portion of the pilot's control or "stick" force that is not linearly correlated with the external forcing function).

It was originally intended that analysis techniques appropriate to sum-of-sines (SOS) inputs would be employed[7]. Because the nominal SOS input used in this study had significant power at off-nominal input frequencies (i.e. "sideband power"), the analysis technique was modified to be appropriate to inputs continuous in frequency. Because of the significant amount of pilot remnant, intra-and inter-subject averaging of the describing functions was performed using techniques recently developed to maximize the reliability of the averaging process[8]. Individual describing functions were not averaged; instead, averaging was performed on the cross-power quantities, first across replications and then across subjects. The inter-subject average describing function was then computed from average cross-power spectra, and the standard errors of the average gain and phase were computed from the variability of the cross-power spectra.

Model Analysis

Because a major objective of the model analysis was to explore the potential of the OCM as a tool for interpolating the results of manned simulations, independent "pilot-related" model parameters were not subjected to an unconstrained identification procedure to obtain the best curve fit possible. Instead, model analysis was initiated with parameters selected from previous laboratory studies, and subsequent parameter changes were made only as required to provide an acceptable subjective match to the data for certain baseline conditions. Predictions were then obtained for other experimental conditions with no further changes in pilot-related parameters.

In order to maximize computational efficiency, most of the model analysis was conducted to provide comparisons with the average performance of the three test pilots. Inter-subject differences were substantial, however, and some analysis was conducted to explore these differences.

In this section of the paper we describe the procedure by which independent model parameters were selected. Interpretation of the specific values selected is provided later.

Non-zero values were assigned to the following pilot-related model parameters:

- Motor Time Constant. This parameter reflects bandwidth limitations imposed by the pilot/stick interface. Values on the order of 0.09 to 0.13 seconds are typical for force sticks and optimal control gains. Larger values have been found for sticks having significant displacement characteristics[9].
- Operator Delay. A pure transport delay is associated with the operator's response characteristics. This OCM independent parameter is generally not influenced by the tracking task; a value of 0.2 seconds is typical. For this study, operator delay was fixed at 0.22 seconds -- 0.20 for the operator and 0.02 for display delay included in the describing function measurements.
- Penalty on Control Force. For tasks in which the control gain is optimal, there is generally no need to assign a performance penalty (i.e., a non-zero "cost coefficient" in the quadratic performance index) to control force or displacement. Where significant forces are required, however, a specific penalty associated with control force (or possibly control rate) improves the match to operator performance[10]. The control penalty is expressed here as the numerical value of the control SD score that contributes one unit of cost to the performance index. (One unit of cost is assigned to an SD score of 1 degree for roll or pitch tracking error.)
- Residual Noise on Error Perception. The OCM allows the user to specify a "residual noise" standard deviation for each of the perceptual variables used by the pilot to reflect the effects of threshold-like phenomena such as perceptual resolution limitations and "indifference thresholds". Residual noise is usually set to zero for laboratory tracking tasks that use symbolic displays with optimal display gains. In general, however, nonzero residual noise should be specified for non-ideal perceptual environments such as real-world visual scene cues[11], or when the operator is indifferent to errors below a certain level.
- Observation Noise Ratio. Pilot remnant is modeled largely as an observation (perceptual) noise process. Except for the residual noise term, observation noise is assumed to scale with the rms value of the perceived variable. An "observation noise ratio" of around -20 dB is typical for laboratory experiments. Based on the results of a recent laboratory study of time-delay effects this parameter was fixed at -19 dB for this analysis[12].
- Motor Noise Ratio. Except for tasks using very large control sensitivities, or human operators with substantial neurological "tremor", this term is negligible. It was fixed at -90 dB for this analysis.
- Internal Motor Noise Ratio. To reflect certain limitations on the operator's ability to predict the effects of his control input on vehicle response, we

assign a small but non-negligible value to the "internal" motor noise (i.e., the pilot's perception of his motor noise variance). This parameter was fixed at -44 dB based on a recent laboratory study[12].

The motor time constant, the control penalty, and residual noise on tracking error were varied during the course of this analysis; the remaining parameters were kept fixed as described above.

For the purposes of model analysis, the "baseline" condition was defined as single-variable tracking, target input, ground-based simulation, zero added delay. There were thus four experimental baseline tracking tasks: pitch and roll for the "F-16" and the "C-141". The approach to model analysis was to first obtain two "calibration" sets of independent model parameters -- one providing the best joint match to "F-16" and "C-141" performance in roll, and one providing a joint match to pitch-axis performance for the baseline conditions. With parameters held fixed at the appropriate baseline values, predictions were then obtained for (1) the effects of additional delay on performance, and (2) performance differences between ground and in-flight simulation.

Values used in the pre-experiment analysis -- shown in the first column of Table 1 -- were used in an initial attempt to match the roll-axis data. These values proved to be too optimistic in terms of predicting average performance. In particular, the experimental tracking errors scores were substantially larger than predicted, the control scores were lower than predicted, and the remnant was greater than predicted.

Table 1: Independent Parameters Adjusted During the Model Analysis

Parameter	Initial	Roll	Pitch
Motor Time Constant, sec	0.12	0.20	0.15
Control for Unit Cost, pounds	0.	0.	20.
Rms Residual Noise, degrees	0.	75.	2.25

The predicted control and control-rate scores were brought into general agreement with the data by increasing the motor time constant to 0.20 seconds. This action had the additional effect of increasing the predicted tracking error score, but not by the amount needed to match experimental results. One or more aditional model parameters had to be modified to provide better matches to experimental tracking error scores and remnant spectra. After some preliminary exploration, we decided to let the residual noise parameter vary.

A residual noise of around 75 degrees was required to provide an acceptable match to the baseline data (i.e., predicted SD scores within 10-15% of the experimental scores) for the roll axis. This value was much larger than would be needed to accommodate an indifference threshold of 5 degrees that one might infer from the instructions to the test pilots. As discussed later, this large value might reflect, in part, significant nonlinearities in the pilot's response behavior.

The pitch-axis baseline tasks were then modeled. The motor time constant of 0.2 seconds obtained in the roll-axis match was reduced to 0.15 to provide a better match to pitch-axis control and control-rate scores, and the residual noise was readjusted to match tracking error. Introduction of the control penalty for the pitch-axis task provided a somewhat better match to the "F-16"/"C-141" performance differences. (A subsequent re-analysis of the roll-axis task showed that the same control penalty had negligible effect on predicted roll-axis performance.)

Once calibrated for the baseline tasks, the pilot-related model parameters considered above were held fixed as the effects of experimental variables were predicted. Changes in experimental conditions were modeled as follows:

- Vehicle delay. The addition of delay to the simulated vehicle's control-system delay was modeled by simply incrementing the "vehicle delay" parameter of the OCM by the same amount. (A baseline vehicle delay of 0.08 seconds was included to reflect irreducible simulator delays exclusive of display-related delay.)
- Ground/flight differences. The parameterization defined above pertains to the ground-based simulation. To account for the effects of in-flight simulation, display variables associated with perception of whole-body motion cues were included in the pilot's "display vector" along with the visual cues of attitude error and error rate. These motor variables were vehicle roll rate and roll acceleration for roll-axis tracking, and vehicle pitch rate, pitch acceleration, and normal acceleration for pitch-axis tracking.
- Combined-axis tracking. Implicit in the parameterization defined above is the assumption that the pilots could devote nearly full attention to the control axis containing the external forcing-function for the single-axis tasks. For combined-axis tracking, we assume that the pilot's must allocate attention in roughly equal proportions to the pitch- and roll-axis tracking tasks. Combined-axis tracking was therefore modeled by increasing the observation noise ratio by 3 dB (equivalent to assigning fractional attentions of 0.5 to all perceived variables)[13].

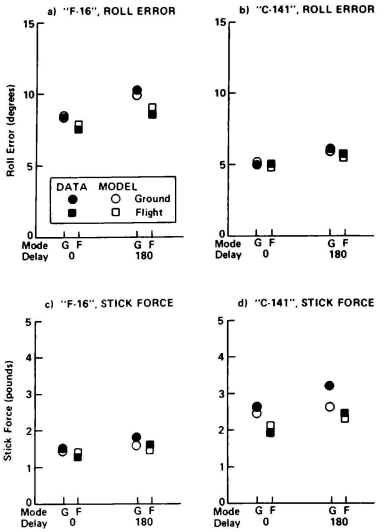


Figure 1: Effects of Delay and Simulator Mode on SD Performance Scores, Roll Axis
Target-tracking task.
Average of 3 subjects, 2 replications/subject.
G = ground-base, F = in-flight

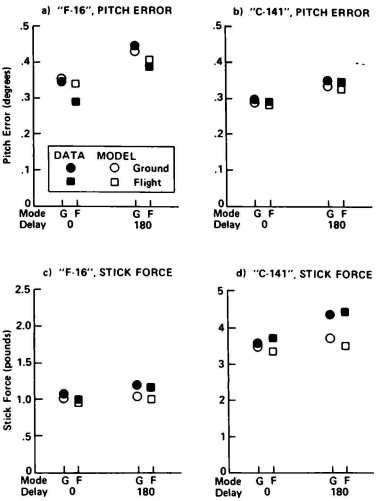


Figure 2: Effects of Delay and Simulator Mode on SD Performance Scores, Pitch Axis
Target-tracking task.
Average of 3 subjects, 2 replications/subject.
G = ground-base, F = in-flight

Principal Results

The major results of the data and model analysis are organized as follows: (1) closed-loop performance metrics, (2) frequency-response measures, and (3) nonlinear analysis of selected time histories. Additional results are provided in Bailey et al.[2]

Closed-Loop Performance Measures

Effects of added delay and simulator mode on measured and predicted SD scores are shown for the roll- and pitch-axis tasks in Figures 1 and 2, respectively. These results pertain to single-axis command-following tasks.

The trends of the tracking error scores were generally as predicted by the pre-experiment model analysis (and confirmed by the post-experiment model analysis shown in the figures):

- Addition of 180 msec delay produced a modest increase in the SD score.
- Delay effects were relatively greater for the "F-16" dynamics than for the "C-141".
- Ground-base simulation produced somewhat larger error scores than in- flight simulation, but the effects of simulator mode on performance were less than the effects of control-system delay for the delays considered in this analysis.

As in a previous study of time-delay effects, the model tended to slightly underestimate the relative increase in error due to added delay[12]. Model/data discrepancies of this sort were, however, within the experimental standard error and cannot be considered statistically significant.

The largest model-data discrepancy concerning tracking error scores -- the only difference likely to be statistically significant -- occurred for "F-16" pitch-axis tracking (Figure 2a), where the model underestimated differences between in-flight and ground simulator performance. This trend has been seen in some previous experiments in which the benefits of whole-body motion cuing have not been fully accounted for by the kind of simple informational treatment conducted here[14].

Ground/flight differences for the remaining error scores ("C-141" pitch, "F-16" and "C-141" roll), however, were predicted rather closely.

Figure 1c and 2c show little measured or predicted effects of experimental variables on the stick SD score for "F-16" tasks. Effects of both delay and simulator mode are apparent in the "C-141" experimental data (Figures 1d and 2d). The model mimicked the noticeable mode difference for the "C-141", zero-delay roll-axis task, but it did not predict the observed effects of delay on the "C-141" stick score.

Table 2 provides a quantitative summary of the relative effects of the experimental factors of control-system delay and simulator mode. Entries under the "delay" heading were obtained by taking the ratio of the population-average error SD score for the 180-msec condition to the average error score for the zero-delay condition. Each entry in the table is the average of four such ratios (pitch and roll, ground and flight). Similarly, entries under the "simulator mode" heading were obtained by averaging the ratios of the "ground" error score to the "flight" error score, over the two axes and the two delay conditions. All data in this table pertain to the single-axis, target-tracking tasks.

Table 2: Effects of Experimental Factors on Average Performance Ratios

	Delay		Simulator Mode	
	"F-16"	"C-141"	"F-16"	"C-141"
Data	1.25	1.21	1.18	1.03
Model	1.19	1.15	1.06	1.06

Averaged over single-axis, target-tracking tasks. Three subjects, two replications/subject.

Table 2 confirms the qualitative impressions obtained from Figure 1 and 2. In terms of the proportional increase in tracking error scores, the addition of 180 msec delay had a greater effect than the transition from flight to ground, and the effects of delay were slightly greater for the "F-16" than for the "C-141". The model replicated these trends, although it did not predict the full extent of the delay effects.

Combined-task error and stick scores for the roll-axis were nearly identical to the single-task scores for the two sets of dynamics, delays, and simulator modes considered in this analysis. Combined-axis/single-axis differences in pitch-axis tracking were less consistent. Error scores for the zero-delay condition were negligibly affected, but the addition of 180 msec delay caused larger performance degradations for the combined-axis tasks. The combined-axis stick scores were 25% to 35% greater than single-axis scores for both dynamics and delay conditions on the pitch axis.

The model overestimated the differences between combined-task and single-task performance for the baseline conditions, predicting differences in the error SD score on the order of 20% to 30% when, in fact, experimental differences were negligible. Because of the clear discrepancy between model and data with regard to taskloading effects, this phase of the model analysis was not continued for the remaining experimental conditions.

The tendency of the model to predict taskloading effects on tracking error when none were found experimentally does not signify a structural defect of the OCM, but rather a deficiency in the method of application. In the discussion section of this paper we suggest a revision in the procedure for selecting pilot parameters that is expected to remedy the overestimation of combined-axis/single-axis differences.

Discussion of the model has concentrated largely on the effects of simulator delay on closed-loop performance. Figures 1 and 2 also verify that the OCM accounted for performance differences between the simulated "F-16" and "C-141" aircraft.

Frequency-Response Measures

Figures 3 through 5 show the effects of three experimental variables (dynamics, delay, and simulator mode) on pilot frequency response. Discrete symbols and smooth curves indicate experimental data and model results, respectively. A gain of 0 dB represents one pound control force per degree display error, and a remnant level of 0 dB represents a control power density level of one pound^2 per radian/second. All results pertain to single-axis, target-tracking tasks.

All frequency-response measurements are shown, regardless of the the standard error. In general, all frequency measurements below 8 rad/sec are "reliable" in terms of a low standard error. The reliability of measurements at higher frequencies fluctuated quite a bit from one condition to the next. In general, measurements at the highest SOS frequency should be regarded with caution.

Figure 3 shows that plant dynamics influenced primarily the pilot's response gain, the phase shift at mid frequencies, and the remnant spectrum in the roll axis. The gain and remnant differences reflect, for the most part, the different control gains provided for the two simulated aircraft. The pitch-axis data (Figure 3b) show different frequency dependencies for the two vehicles as well as differences in overall gain and remnant levels. In particular, the "F-16" dynamics showed a more pronounced high-frequency peak in the gain response and a flatter phase response up to about 10 rad/sec than the "C-141" dynamics. Differences in pilot response across simulated vehicle and across axes of control were mimicked by the model.

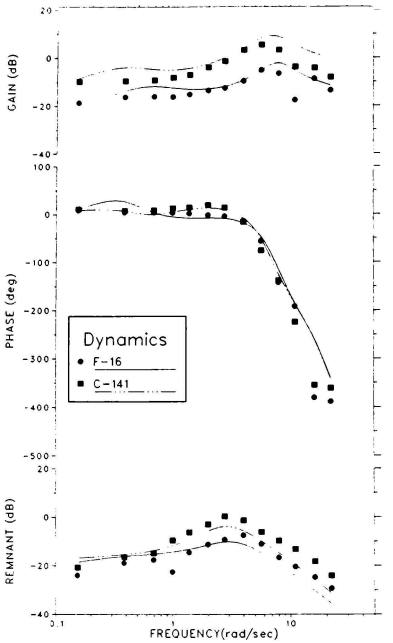


Figure 3a: Effects of Dynamics on Pilot Frequency Response.
Average of 3 subjects, 2 replications/subject.
Single-axis, zero-delay, ground-based, target-tracking Task.

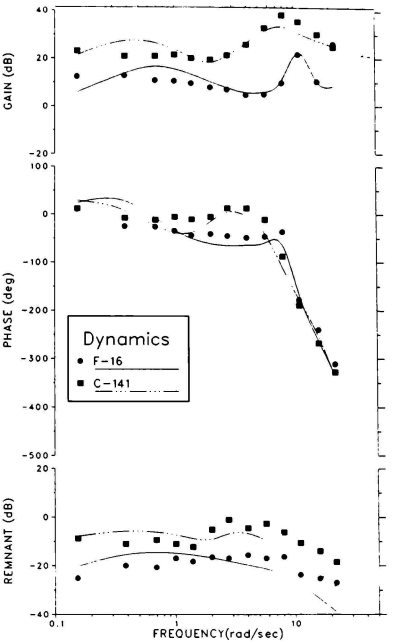


Figure 3b: Effects of dynamics on Pilot Frequency Response, Pitch Axis
Average of 3 subjects, 2 replications/subject.
Single-axis, zero-delay, ground-based, target tracking task.

As might be expected from the preceding discussion of closed-loop performance metrics, the effects of delay and simulation mode on pilot frequency response were substantially less than the effects of simulated aircraft dynamics. Figure 4 shows that the qualitative effects of adding delay, were to (1) slightly decrease pilot gain at low and mid frequencies, (2) decrease the frequency at which the gain curve peaks, (3) decrease phase lag (or increase phase lead) at mid frequencies, and (4) increase pilot remnant. As correctly predicted by the model, the effects of delay on gain and phase shift were more pronounced for the pitch axis response.

Figure 5 shows some reduction in remnant and phase lag with the transition from ground to flight. The remnant curve may be interpreted as indicating a slightly wider response bandwidth for the in-flight tasks. The only sizeable effect predicted by the model (and not revealed in the data) was a mode-related difference in phase shift at the lowest measurement frequencies.

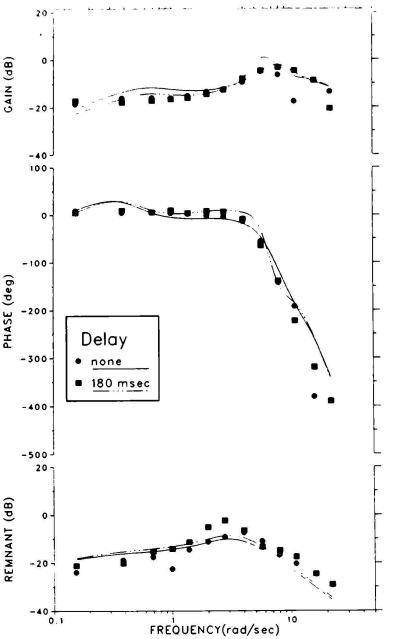


Figure 4a: Effects of Delay on Pilot Frequency Response, Roll Axis
Average of 3 subjects, 2 replications/subject.
Single-axis, F-16, ground-based,
target-tracking task.

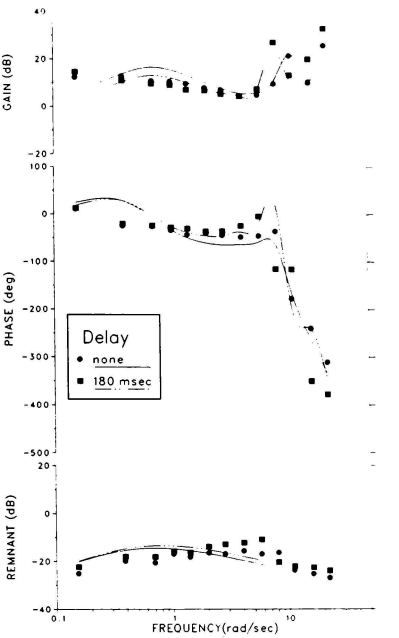


Figure 4b: Effects of Delay on Pilot Frequency Response, Pitch Axis
Average of 3 subjects, 2 replications/subject.
Single-axis, F-16, ground-based,
target-tracking task.

Nonlinear Analysis

The exploration of potential nonlinearities in the operator's response behavior was motivated by the unusually large values for the "motor time constant" and "residual noise" parameters needed to provide an acceptable match to the average performance of the three test pilots. Examination of the individual performance of the three test pilots showed considerable differences between the best-performing pilot (in terms of minimizing rms roll error) and the other two pilots. These results prompted the suspicion that the response strategies of the two pilots who performed less well might have contained significant nonlinearities, and that these nonlinearities might have resulted in the apparent reduced bandwidth and larger tracking error characteristic of these pilots.

Time histories of one trial each for the three test pilots were examined for evidence of nonlinear response behavior. The experimental condition represented by these trials was roll-axis tracking, ground-based simulation, "F-16" dynamics without additional time delay, with an external sum-of-sines target-command signal active in the roll axis only.

Time histories generated by the best-performing and worst-performing pilots were inspected. Numerical

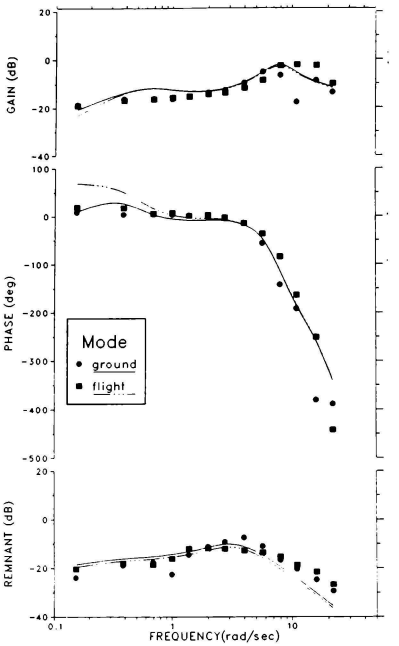


Figure 5a: Effects of Simulator Mode on Pilot Frequency Response, Roll Axis
Average of 3 subjects, 2 replications/subject.
Single-axis, F-16, zero-delay,
target-tracking task.

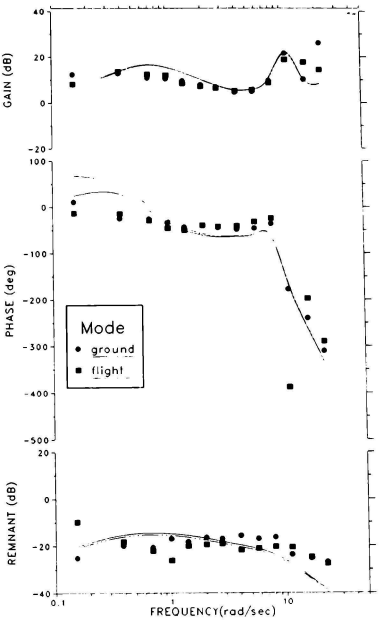


Figure 5b: Effects of Simulator Mode on Pilot Frequency Response, Pitch Axis
Average of 3 subjects, 2 replications/subject.
Single-axis, F-16, zero-delay,
target-tracking task.

printouts of the stick time histories were initially scanned for the appearances of intervals of little or no stick motion ("flat spots"). Five-second intervals containing these occurrences were then plotted for inspection.

In order to provide references against which to compare the experimental time histories, the following model analysis was performed. First, the various performance metrics obtained from the two experimental trials were matched with the steady-state optimal control model (OCM). A "simulation" implementation of the OCM[15] was then used to generate stick time histories for the 5-second time segments of interest, using the appropriate Calspan experimental forcing function as the driving function. If the model were to match the pilot's linear characteristics perfectly, and if the effects of pilot remnant were negligible, the model and experimental stick time histories would be identical. Given the existence of remnant and the difficulty of a perfect match, the model-generated time histories are intended more as a qualitative reference. Specifically, one can determine whether or not the model predicts the response trends exhibited by the data.

Time histories for the worst-performing pilot are shown in Figure 6. Four plots are shown: (a) the experimental stick time history, (b) the stick time history predicted by the model, (c) the experimental error time

history, and (d) the predicted error time history. Because the model analysis and experimental simulation used different sign conventions, the predicted error waveform shown in Figure 6d has been inverted to facilitate comparison with the other curves shown in the Figure.

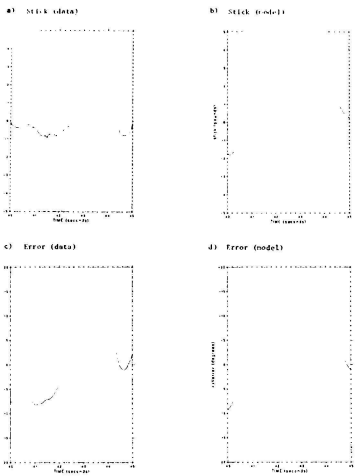


Figure 6: Time Histories for Pilot B

Intermittent nonlinear response behavior is suggested by the experimental "stick" time history of Figure 6a which shows a relative pause in response activity of about 1 second, beginning around 43.5 seconds into the run. The error signal plotted below does not reveal a similar flat spot, but rather a local maximum as the error reverses direction and heads toward zero. Furthermore, the predicted stick response shows a more peaked response pattern (Figure 6b) in response to the same type of error pattern (Figure 6d) than is found in the data. The experimental stick time history suggests a "move-and-wait" strategy in which the pilot maintains a relatively constant control force that is sufficient to cause the error to begin to decrease. Other segments of this pilot's response time history (not shown here) revealed qualitatively similar patterns, whereas similar patterns were not found in the stick time history for the best-performing pilot.

Because of the limited database explored, conclusions drawn from this subjective analysis must be considered tentative. The results suggest the following trends:

1. Better-performing subjects, in terms of rms error minimization, tend to adopt a response strategy that is more nearly linear than subjects who perform less well.
2. Nonlinearities present in the operator's response strategy include move-and-wait episodes during which the operator maintains a relatively constant control force.

Additional study is needed to determine the extent to which these tentative conclusions can be generalized.

Summary and Discussion

For the conditions analyzed by BBN, the trends predicted by pre-experiment model analysis were largely confirmed by the experimental study and replicated by post-experiment model analysis. Specifically,

- Addition of 180 msec delay to the flight-control system caused a modest increase (around 22%) in rms tracking error.
- Delay had a larger effect on performance with the simulated "F-16" than with the simulated "C-141".
- Error scores were, on the average, larger in the ground-based simulations than in the in-flight simulations, but ground/flight differences were relatively small (about 10% on the average).
- Performance trends were similar for the pitch and roll axes.
- The model, "calibrated" for the reference conditions in pitch and roll, replicated the important performance trends, but tended to somewhat underestimate the quantitative effects of delay and simulator mode.

The reader should note that all model results shown in this memorandum were obtained with but two sets of model parameters: one set adjusted to jointly match roll-axis performance for the "F-16" and "C-141" in the single-axis, ground-based, zero-delay, target-tracking task, and one set adjusted to match pitch-axis performance.

The major discrepancy between experimental and model results concerned the differences between single-task and combined-task performance, where the predicted differences were substantially greater than those observed in most cases. As suggested earlier, we suspect this discrepancy was due largely to an error in model parameterization. Recall that the "single-axis" tasks required the pilot to control the airplane in both pitch and roll in response to an external target input in only one axis. We modeled this situation by (1) selecting observation noise parameters appropriate to true single-axis tracking (i.e., aircraft motion allowed in only one

axis), and then (2) accommodating the combined-axis task by doubling the observation noise/signal ratio to reflect an even split of attention between the two axes. With the advantage of hindsight, a better strategy might be to assume that the pilot must share attention in a near-equal manner between the two axes whenever the two axes must be controlled, even if there is an explicit input in only one axis.

It seems clear that selecting observation noise/signal ratios appropriate to combined-axis tracking for the so-called "single-axis" tasks would have allowed us to match the experimental results with "residual noise" levels closer to what would be associated with reasonable indifference thresholds. (Contract resources did not permit a repeat of the model analysis to test this hypothesis.)

On the basis of the results to date, we tentatively conclude that the OCM, properly calibrated, has the potential for interpolating the results of manned simulation studies for the type of steady-state flight tasks explored in the Calspan experimental study. Although we have found the OCM to forecast performance trends with independent parameters selected entirely on the basis of previous results, we expect model application to be most reliable when the model is applied in conjunction with a manned simulation study, with parameters selected to match key reference conditions. Calibration against the specific data base will allow consideration of factors specific to the simulation study, such as (1) the performance goals, motivation levels, and general piloting techniques of the specific test pilots: (2) the nature and amount of task-specific training received by the test pilots, and (3) other factors that account for the difference between laboratory tracking studies using over-trained college or high-school students and simulation studies using actual pilots performing representative flight tasks under constraints often avoided in the laboratory.

Tentative rules for calibrating the model are:

- Calibrate the model against the extreme conditions explored in the simulation study (e.g., widest- and narrowest-band vehicle dynamics, smallest and largest delay common to all vehicles, etc.).
- Adopt attention-related parameters appropriate to the number of axes controlled -- not to the number of axes containing external inputs.
- Where control-gains are not optimized for aggressive operations, test for the need of a specific penalty on control force to model the pilot's reluctance to generate large average control forces.
- Select residual noise parameters to reflect, at a minimum, perceptual resolution limitations and the pilot's lack of concern about reducing errors below some acceptable minimum.
- In general, follow the model-matching procedure described in this paper.

References

1. Bailey, R.E., Knotts, L.H., Horowitz, S., and Malone, P.,"Effect of Time Delay on Manual Flight Control and Flying Qualities During In- Flight and Ground Based Simulation", Proc. AIAA Flight Simulation Technology Conference, Monterey, CA, August 17-19, 1987.
2. Bailey, R.E., et al., (Calspan report in preparation).
3. Kleinman, D.L., Baron, S., and Levison, W.H., "An Optimal-Control Model of Human Response, Part I: Theory and Validation", Automatica, Vol. 6, pp. 357-369, 1970.
4. Kleinman, D.L., Baron, S., and Levison, W.H., "A Control Theoretic Approach to Manned-Vehicle Systems Analysis", IEEE Trans. on Auto. Control, Vol. AC-16, pp. 824-833, No. 6, December 1971.
5. Levison, W.H., "The Optimal Control Model for the Human Operator: Theory, Validation, and Application," in Frazier, M.L., and Cromble, R.B., (eds): Proceedings of the Workshop on Flight Testing to Identify Pilot Workload and Pilot Dynamics, AFFTC-TR-82-5, Air Force Flight Test Center, Edwards Air Force Base, CA, May 1982.
6. Levison, W.H., "Application of the Optimal Control Model to the Design of Flight Simulation Experiments," SP-634: Flight Simulation/Simulators, Society of Automotive Engineers Paper No. 851903, Aerspace Technology Conference and Exposition, CA, October 1985.
7. Zacharias, G.L. and W.H. Levison, "A Performance Analyzer for Identifying Changes in Human Operator Tracking Strategies", AMRL-TR- 79-17, Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, March 1979.
8. Levison, W.H., "Some computational Techniques for Estimating Human Operator Describing Functions", Proc. of the Twenty-first Annual Conference on Manual Control, NASA Conference Publication 2428, pp. 31.1-31.21 , May 1986.
9. Repperger, D.W, and Levison, W.H., "Effects of Control Stick Parameters on Human Controller Response", Proc. of the Twentieth Annual Conference on Manual Control, Nasa Conference Publication 2341, Volume I, pp. 157-166, June 1984.
10. Levison, W.H., "Effects of Control Stick Parameters on Human controller Response", Report No. 5510, Bolt Beranek and Newman Inc., Cambridge, MA, January 1984.
11. Levison, W.H. and Warren, R., "Use of Linear Perspective Scene Cues in a Simulated Height Regulation Task," Proc. of the Twentieth Annual Conference on Manual Control, Nasa Conference Publication 2341, Volume I, pp. 467-490, June 1984.

12. Levison, W.H. and A.W.F. Huggins, "Modeling the Effects of Delays and Lags on Tracking Performance", Report number 6245, BBN Laboratories Incorporated, Cambridge, MA, June 1986.

13. Levison, W.H., "A Model for Mental Workload in Tasks Requiring Continuous Information Processing," in Moray, N. (ed): Mental Workload Its Theory and Measurement, Plenum Press, New York and London, 1979.

14. Levison, W.H., Lancraft, R.E., and Junker, A.M., "Effects of Simulator Delays on Performance and Learning in a Roll-Axis Tracking Task (Formal)", Technical Report AFFDL-TR-79-3134, Proc. of the Fifteenth Annual Conference on Manual Control, November 1979. Wright-Patterson Air Force Base, Ohio.

15. Kleinman, D.L., Baron, S., and Berliner, J.E., "MCARLO: a Computer Program for Generating Monte-Carlo Trajectories in a Time-Varying Man/Machine Control Task", Technical Report TD-CR-77-2, U.S. Army Missile Research and Development Command, Redstone Arsenal, Alabama, June 1977.

87-2372

TEMPORAL FIDELITY IN AIRCRAFT SIMULATOR VISUAL SYSTEMS

Michael S. Merriken
Systems Research Laboratories, Inc.
Dayton, OH 45440

Gary E. Riccio
Armstrong Aerospace Medical Research Laboratory
Wright-Patterson Air Force Base
Dayton, OH 45440

William V. Johnson
Systems Research Laboratories, Inc.
Dayton, OH 45440

Abstract

This paper will discuss a research effort of the Armstrong Aerospace Medical Research Laboratory to investigate the effects of temporal fidelity in aircraft simulator visual systems. A review of the pertinent previous research is presented first followed by an overview of the research approach taken by the Human Engineering Division. A summary of the completed research to date, with the salient results, is then presented. Finally, the program plan for the next phase of research is discussed.

Introduction

Temporal fidelity in aircraft simulators is a critical issue for training effectiveness and aeronautical research and development. The accuracy of the system's response can mean the difference between effective and ineffective training or between an acceptable or unacceptable rating of handling qualities for a new aircraft configuration. The Human Engineering Division of the Armstrong Aerospace Medical Research Laboratories located at the Wright-Patterson Air Force Base is currently engaged in a research effort to investigate requirements for effective flight simulator displays.[10] An element of this effort is to quantify the effects of temporal delay, focusing on the acquisition of flight-control skills, in aircraft simulation.

Aircraft simulators offer several advantages over actual flight experience. Maximum safety, continuous availability, high utilization, and low operating costs are the operating advantages of flight simulators. The training advantages of simulators are absolute control of the training environment, standardization of training, automated training, and practice in otherwise impractical tasks.[11] Thus, simulators have demonstrated their value and will continue to be used as a tool by the aeronautical community.

Simulation by definition means "an imitative representation of the functioning of a system".[20] Today's simulators have visual systems that attempt to mimic the actual flight environment. The combination of the computational requirements of flight simulation and the current state of computer hardware is such that there are limits to what characteristics of the flight environment the simulator can reproduce. Additionally, the computation of the aircraft states and the implementation of the corresponding visual display result in delayed feedback about the pilot control actions and the displayed aircraft states. These factors result in trade-offs between visual fidelity and temporal fidelity.

The fidelity of a simulation influences flight-control performance in the simulator and transfer of training to the real aircraft. The level of performance achieved across a range of time delays is a useful metric for engineering and design applications. Transfer of training is a useful metric for the training value of a simulator.

Summary of Previous Research

A review of the literature reveals that the effects of visual feedback delays on performance vary with one or any combination of the following variables; axis of control, aircraft dynamics, task type and difficulty, and the availability of whole-body motion information.[12]

Visual delays seem to affect the control axes differently. Degradation of roll axis control because of delayed visual scene response can be more severe than effects on pitch axis control.[4,16] This is consistent with other data on axis of control[3,14] This may be due, in part, to greater system bandwidth of the roll axis. Previous control research has demonstrated that the bandwidth (i.e. responsiveness) of the aircraft influences the effect of time delay on pilot performance[2,15].

Another study reported that the human system bandwidth decreased as system time delay increased.[1]

The influence of aircraft dynamics is complex. Ricard and Puig[15] concluded that the control of both sluggish and highly-responsive aircraft is very sensitive to visual feedback displays. Queijo and Riley[17] manipulated the longitudinal short-period characteristics of a fixed-base aircraft simulator and demonstrated a positive correlation between rated handling qualities and amount of acceptable time delay. Miller and Riley [13] added motion to the experimental paradigm and found that the inclusion of relatively complete motion cues enabled pilots to maintain acceptable performance for longer time delays. The number of degrees of freedom of motion employed was a significant factor.

Increasing the difficulty of the flight task typically exacerbates the problems associated with time delay. The results of Miller and Riley[13], Queijo and Riley[17], and Sevier et. al. [18] support this thesis. The results of Cooper, Harris and Sharkey [3], though, provide data to the contrary. It seems reasonable to assume that the particular task and the method by which the difficulty is increased (e.g. varying gust frequency and amplitude, demand of measurement criteria, initial condition, etc.) will influence the effects of this manipulation.

Whole body motion simulators add an additional complicating factor to this situation. The large mass of the motion base results in a sluggish response to control input. Typically, the delay of the motion system is different than the delay of the visual system and results in a temporal mismatch of information. The effects of mismatch on transfer of flight training are virtually unknown.[8] Gum and Albert[6] reported that Air Force instructor pilots preferred the visual system to be as fast as possible regardless of the disparity between the visual and the motion cues. Frank, Casali, and Wierwille[5] report that visual delay is more detrimental to operator control performance than is motion delay. They recommend that visual system movement lead motion system movement.

Interestingly, the Federal Aviation Administration requirements for advanced simulation training seem to contradict these findings.[19] They state that "Visual scene changes from steady state disturbance shall not occur before the resultant motion onset but within the system dynamic response tolerance of 150 ms". Part of the problem is that the method for measuring delays is not specified, although the document implies that visual scene position responses shall not lead motion base acceleration responses to a step input. This makes it difficult to interpret and follow the guidelines. This disparity of recommendations will be addressed in future research in this laboratory.

Approach at AAMRL/Human Engineering Division

To study the effects of visual-system time delay on the acquisition of flight control skills, the research program was designed to maximize comparability across experiments. The first studies were exploratory in nature and provided indications about the strength of various effects of visual delays on operator performance and on the control of the simulated flight. The results of these experiments provided guidelines about the experimental designs (e.g. the number of subjects, the number and range of time delays, the amount and type of training required) and the simulation equipment (graphic systems). These results also provided insight into the operator performance strategies through frequency analysis of control inputs and through evaluation of transfer of training.

Transfer of training is operationally defined as the effect of experience in a particular task on performance in a different task. The extent of transfer between simulators cannot yet be predicted from the respective patterns of training performance (proficiency or style). The research is addressing the usefulness of the training received in a simulator with poor temporal fidelity for subsequent performance in a simulator with high temporal fidelity (i.e. with less delay).

The method of control used by the subjects is also of interest. It is accepted that time delays decrease the stability of the pilot-aircraft system. Open-loop and human-operator describing functions are being utilized to characterize subject behavior in each study. Analysis of the variations of cross-over frequency, phase margin, gain, and non-linear remnant across time delays will provide an indication of the effects on vehicle control. These analysis techniques will be consistent throughout the research effort and will provide for comparisons between studies.

An approach to ameliorate the

effects of time delay will also be evaluated. To date, linear and non-linear delay compensation approaches have demonstrated some success but typically possess limitations regarding implementation.[2,6,7,12] Exploratory studies will examine the use of supplementary cues (e.g. peripheral visual displays, g-seat) in the control of aircraft altitude and heading. The supplementary cues may be more resistant to delay effects if presented to rate-sensitive perception systems such as the peripheral retina and the tactual/kinesthetic systems. This research will investigate whether supplementary motion information should be presented with a minimum time delay or should be synchronized with task-critical information in a slower central display. Transfer of training data will indicate whether flight-control skills are depenent on the synchrony of information.

Delay Verification

Implicit during the study of temporal fidelity is the verification of the system time delay. This is paramount when making recommendations that can be followed by the simulation community. Frequency-domain techniques have been used for verification for these experiments since it measures any combination of dynamic and pure delays and can be easily replicated.

To measure the transport delay, several test frequencies were substituted for stick commands inputs. A photocell was used to measure the differences in display luminance. The phase difference between the input to the aircraft dynamics and the output measured by the photocell was determined by a frequency analyzer (Bafco model 916). The phase lag due to the aircraft dynamics was subtracted from the measured phase difference at each of the test frequencies. The transport delay was then calculated by dividing the adjusted phase difference by the product of the test frequency and one revolution.

Summary of Completed Experiments

The experiments during this initial effort employed single axis and dual axis altitude and roll maintenance tasks. Aircraft dynamics were modeled using first and second-order transfer functions and the scene graphics were highly schematic. Each new experiment was a logical extension of the previous studies. This phase was recently completed and the reader is directed to Fisher et. al.[4] and Riccio et. al.[16] for detailed information.

The results of these early experiments began to show trends in training performance and transfer of training with respect to the effects of transport delays. The effects of delay consistently were greater for roll axis control than for pitch axis control. For delays less than 100 ms, the results indicated that subjects fully compensated for the delayed feedback by reducing phase lag. Therefore, delays in this range may be acceptable for simulations and flight control systems performing disturbance regulation tasks. The initial results also indicated that for subjects trained with visual system delays up to 200 ms, there may be no important effects on transfer of training to a 50 ms delay. A consistent trend, though not statistically significant, was that visual system delays on the order of 100 ms may actually facilitate transfer of training.[16] This result will be investigated further in later studies.

Levison and Papazian[9] analyzed data, from recent experiments conducted by Arvin/Calspan Corporation with the NT-33 in-flight simulator, on the effects of time delay on flight performance.[9] Large transport and fighter dynamics were simulated. Ground-based and in-flight conditions involving target following or disturbance regulation tasks with explicit roll and pitch references were presented on a head-up display (HUD). Delays of zero and 180 ms were added to the system delay. Only three subjects participated in this experiment but the results indicated that there was a modest increase (20%) in tracking error score with the larger delay, relatively small differences (less than 10%) in performance existed between the in-flight and ground-based simulations, and the relative effects of delay were similar for both pitch and roll. This last result conflicts with the previously mentioned studies. Further research is planned with the NT-33 as well as comparisons with the fixed-base simulators.

Planned Experimentation

Follow-on experiments will concentrate on evaluating the effects of temporal delays on operationally realistic flying tasks. Both fighter-type and transport-type dynamics will be utilized. These aircraft types will be modeled using a six degree-of-freedom, stability derivative based model. This will provide a higher degree of fidelity of aircraft responsiveness. A large-screen projection system will be incorporated in the new simulations to provide a larger field of view. Data will be

collected to continue the analysis of training performance, transfer of training, and operator control techniques.

The operational tasks that will be used include a side-step landing task, a terrain following/terrain avoidance tasks, a low altitude parachute extraction task, and multi-aircraft tasks (formation flying and air combat). Experiments will also address: (1) the trade-off between time delay and the display complexity; and (2) delay compensation with the use of supplementary displays.

Air Force pilots will be included in these experiments in addition to the continued use of non-pilots as the primary source of subjects. Additionally, motion base effects will be included in some of these studies both from motion-base simulators and the NT-33.

REFERENCES

1. Allen, R. W. and DiMarco, R. J. (June, 1984). Effects of transport delays on manual control system performance. In the Proceedings of the 20th Annual Conference-on Manual Control (pp 185 - 210). NASA Ames Research Center; Moffett Field, CA.

2. Crane, D. F. (1983). Compensation for time delay in flight simulator visual display systems. In Proceedings of the AIAA Flight Simulation Technologies Conference. Paper no. 83-1080, Niagra Falls, NY.

3. Cooper, F. R., Harris, W. T., and Sharkey, V. J. (1975). The effect of delay in the presentation of visual information on pilot performance. (NAVTRAEQUIPCEN IH-250). Orlando, FL: Naval Training Equipment Center.

4. Fisher, T. J., Riccio, G. E., and McMillan, G. R. (1986). The effects of simulator delays on the acquisition of flight control skills. In Proceeding of the 10th Biennial Psychology in the Department of Defense Symposium (USAFA-TR-86-1). Colorado Springs, CO: U.S. Air Force Academy.

5. Frank, L. H., Casali, J. G., and Wierwille, W. W. (1986). Effects of visual display and motion system delays on operator performance and uneasiness in a driving simulator. Unpublished doctoral dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA.

6. Gum, D. R. and Albery, W. B. (1976). Time-delay problems encountered in integrating the advanced simulator for undergraduate pilot training. AIAA Journal of Aircraft; Vol. 14, No. 4, pp. 327-332.

7. Hess, R. A. and Myers, A. A. (1984). A nonlinear filter for compensating for time delays in manual control systems. In Proceedings of the 20th Annual Conference on Manual Control, NASA/Ames Research Center, CA.

8. Levison, W. J., Lancraft, R. E., and Junker, A. M. (1979). Effects of simulator delays on performance and learning in a roll-axis tracking task. Proceeding of the 15th Annual Conference on Manual Control(AFFDL-TR-79-3134). Wright-Patterson Air Force Base: Air Force Flight Dynamics Laboratory.

9. Levison, W. H. and Papazian, B. (April, 1987). Post-experimental model analysis in support of the NT-33 ground simulator time delay study. BBN Laboratories, Inc. Technical Memorandum EPD-87-1. Cambridge, MA.

10. Martin, E. A., McMillan, G. R., Warren, R., and Riccio, G. E. (May, 1986). A program to investigate requirements for effective flight simulator displays. In the Proceedings of the International Conference on Advances in Flight Simulation Visual and Motion Systems. The Royal Aeronautical Society; London, England.

11. McLanaghan, R. (1986). Flight simulators-purpose and architecture. In Flight Simulator Update - 1986. University Center at Binghamton, NY.

12. McMillan, G. R. (1986). Cue integration and synchronizaiton. Flight Simulation Update 1986. University Center at Binghamton, NY.

13. Miller, G. K. and Riley, D. R. (1976). The effect of visual-motion time delays on pilot performance in a pursuit tracking task. In the proceedings of the AIAA Visual and Motion Simulation Conference (pp. 55-62). Dayton, OH.

14. Ricard, G. L., Norman, D. A., and Collyer, S. C. (1976). Compensating for flight simulator CGI system delays. In Proceedings of the 9th NTEC/Industry Conference (NAVTRAEQUIPCEN IH-276). Orlando, FL: Naval Training Equipment Center.

15. Ricard, G. L. and Puig, J. A. (March, 1977). Delay of visual feedback in aircraft simulators (NAVTRAEQUIPCEN TN-56). Naval Training Equipment Center, Orlando, FL: Human Factors Laboratory.

16. Riccio, G. E., Cress, J., and Johnson, W. V. (1987). The effects of simulator delays on the acquisition of flight control skills: control of heading and altitude. Manuscript submitted for publication.

17. Queijo, M. J. and Riley, D. R. (1975). Fixed-base simulator study of the effect of time delay in visual cues on pilot tracking performance. NASA Technical Note D-8001. Langley Research Center, VA.

18. Sevier, J. A., Minturn, D. B., Bernard, D. W., and Pollard, T. J. (1984). The effect of computational time delays on pilot performance in real-time flight simulation. In Proceedings of the 22nd Aerospace Sciences Meeting (No. 84-0553). Reno, NV.

19. U. S. Department of Transportation (1980). Federal Aviation Administration; Advanced Simulation. Federal Register, Part V; Vol. 45, No. 127, June 30, 1980.

20. G. and C. Merriam Company (1977). Webster's new collegiate dictionary. G. and C. Merriam Co, Springfield, MA.

87-2373

The Impact of Network Delay on Two-Ship Air-to-Air Combat Simulation

Capt Hugh L. (Pat) Malone III
Capt Scott Horowitz, PhD
Capt John A. Brunderman
Air Force Human Resources Laboratory
Williams AFB, AZ

Herbert Eulenbach, Exchange Engineer, MOD
Federal Republic of Germany

Abstract

The Air Force is currently developing the capability to network aircraft simulators via communications links (WARNET) to provide more realistic air combat training at the squadron level. The Army recently developed the same capability with tank simulators (SIMNET). One of the factors that affects the training effectiveness and realism of these devices is the transport delay of the system. The transport delays can be broken down into two major categories: (1) within-simulator - the transport delay of the individual simulator, and (2) between-simulator - the transport delay between simulators due to communications time and interfacing. The between-simulator transport delay (BSTD) must be considered in order to determine the limitations of a combat air-to-air trainer in a simulator network. This work investigates the problem of BSTD on a two-ship engagement using Air Combat Maneuvering Instrumentation (ACMI) data. ACMI data provides six degrees of freedom information on actual air-to-air engagements. The ACMI data was replayed on an Integrated Raster Imaging System (IRIS) which provided images of the engagement as seen out-the-window from the attacker's cockpit. The six degrees of freedom data for each aircraft can be replayed in real time or distorted (delayed) in time by the use of simple algorithms. The images provided by the IRIS were frozen at times of interest during the engagement allowing an assessment of the aircrafts' relative positions using pilot evaluations. After the pure delays were tested the use of a simple first order predictor was investigated. The results of this experiment provide the designers of networked simulators a guide as to the amount of BSTD that can be tolerated and the effects of prediction algorithms.

Introduction

The objective of this study was to define the maximum tolerable transport delay between simulators, the between-simulator transport delay (BSTD). The definition as to the maximum transport delay between networked simulators is an important design point for providing high fidelity combat training. In rapidly changing air-to-air (A/A) flight environments, relative position/-attitude of all participants is the basis for the pilot's choice of maneuver and implementation of ordinance. The perceived position of one aircraft to another can be greatly affected by a delay in receiving information as to the actual position of the other aircraft. A delay in an aircraft's simulated flight path results in the attacking pilot changing his tactics in order to better position himself. When a pilot changes his tactics in the simulator versus what he would do in the aircraft, questions as to the effectiveness of the training arise. If the limit as to the maximum delay can be defined, a network can be designed with transmission delays so as to ensure transmission delays below this maximum threshold are achieved.

In order to describe the approach taken, it is first convenient to point out the limitations of this study and define a few terms. It is obvious that the highly dynamic environment of an A/A engagement is virtually impossible to model completely. With this in mind, the basic effect of BSTD is depicted by the scenario in Figure 1. In the simulator, the attacker (A) would see the defender (D) delayed from a point in time (N) by some time (τ).

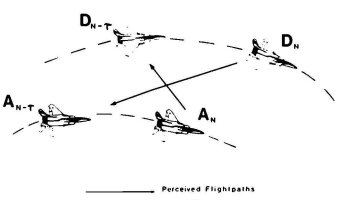


Figure 1. Perceived Flight Paths due to BSTD.

The same is true for the defender viewing a delayed attackers' position. In an actual simulated sortie, the pilot of "A" would make flight path corrections relative to the defender's delayed image and vice versa. These changes are highly dynamic and are totally pilot dependent; i.e., the pilot may choose a different tactic given the same

Chief Engineering Evaluation Pilot
Member AIAA

relative geometry. This limits the fidelity as to how well this scenario can be modeled. However, due to the delayed images, simulator pilots will make corrections to a perceived flight path - that of the delayed image. The recorded Air Combat Maneuvering Instrumentation (ACMI) data allows us a means of evaluating changes to the actual or delayed flight paths. We approached the question as to the maximum tolerable BSTD by finding the maximum BSTD that could be added before a pilot would change his tactics. The question is: how much BSTD can be added to the viewed aircraft's flight path before a change in tactics will occur? The delay which causes a change in tactics is defined, for the purpose of this experiment, as the maximum tolerable between-simulator transport delay.

Experimental Set-Up

In investigating the effects of BSTD on A/A engagements, we were faced with a flight environment in which no two engagements would be the same. Knowing this fact, we attempted to determine the effects of BSTD on somewhat "generic" flight maneuvers. This allowed us to investigate typical maneuvers flown in an A/A engagement. A basic fighter maneuver (BFM) sortie was flown using two F-15A aircraft on the ACMI range at Luke AFB, AZ. A review of the sortie was conducted by a highly experienced air-to-air (A/A) flight instructor and four maneuvers were selected. The maneuvers cover those most commonly used in an A/A engagement: A gun tracking shot, missile shot, high deflection gun shot, and a defensive maneuver.

The ACMI range data was reduced and replayed on a real-time Integrated Imagining Raster System (IRIS). The IRIS generated a real-time, three dimensional dynamic display of each maneuver. An out-the-window view from the attacker's aircraft was used. The view from the defender's aircraft is not available from IRIS in real time. The view from the attacker's aircraft allows one to view the defending aircraft's flight path in real-time. The ACMI data provides vector positions of the aircraft in space allowing a means for delaying the flight path of the defending aircraft, as displayed on the IRIS, for a series of predetermined delays.

To truly investigate the transport delay effects using a computer model requires the use of a model of the figher pilot as well. Since this experiment is a preliminary investigation to determine "tolerable" transport delays in a networked system, the numerical scheme described below was utilized. The ACMI data was recorded at 10Hz thus providing new position data every .10 seconds. Figure 2 shows the effects of delay (τ) on the relative position vectors (A, D) of the attacker and defender with no pilot compensation. The subscripts N correspond to the instantaneous real-time data. If the information transfer is delayed by 2 frames (.20 secs) then the following occurs: The defender sees the attacker at time N-2 while the attacker sees the defender at time N-2.

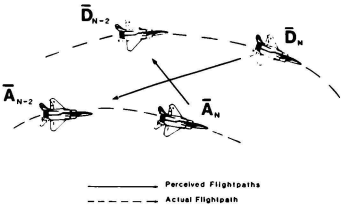


Figure 2. Relative Vector Positions due to BSTD

The major disadvantage with this scheme is the distortion of the relative geometry due to lack of pilot compensation. To reduce this effect a type of center differencing scheme is utilized. This method better preserves the relative positions of the aircraft while showing delay effects, as shown in Figure 3. Using this approach for the delay algorithms, we were able to display the engagements and introduce various between-simulator delays. The use of a predictor to decrease the BSTD effects was also studied using a simple first order predictor based on Figure 3.

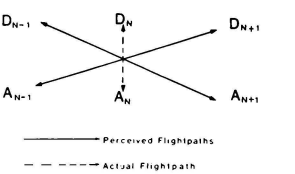


Figure 3. Relative views for simulator scenes due to BSTD

The gun tracking shot, high deflection gun shot, missile shot, and defensive maneuver were then displayed on the IRIS with and without the predictor. The maneuvers were shown as an out-the-window dynamic scene from the attacker's cockpit. BSTD delays of 0, 0.2, 0.5, 1.0, and 1.5 seconds were introduced to the display. The dynamic display was split screen allowing the same maneuver to be displayed real-time using various delay pairs. This permitted a realistic scene in which a direct comparison of engagements could be made. The delay pairs were set up so that each delay case appeared an equal number of times and all possible delay pairs were investigated. This resulted in an array in which each maneuver was shown ten times each in two sets - one with and one without the predictor. The entire display array was recorded on VHS in order to ensure each experimental run would be the same for each review. Five highly experienced fighter pilots with an

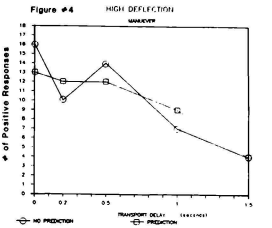
average of 1500 hours of fighter time reviewed the displays. The pilots were unaware as to which time delay would be used and the presentation of delays was highly randomized. Each pilot then chose which of the two displays provided the best shot or in the case of the defensive maneuver, permitted the best defensive maneuver.

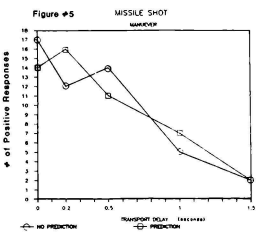
Results

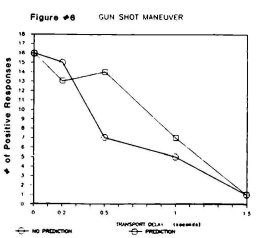
The desired result was to determine the maximum tolerable BSTD that could be accepted before a pilot would make a change in tactics. The approach taken was to provide curves which depict the degradation of an A/A to engagement due to BSTD. This was accomplished under the premise that the BSTD which causes a significant degradation in the engagement, is the BSTD when the pilot would change tactics. These curves were generated using pilot evaluations and the results are shown in Figures 4-7. Each curve represents the number of times the pilot chose a display as providing the best shot/defensive maneuver. The curves represent an expected trend, as the transport delay increased, the pilots rated the maneuver as adequate less often. Each maneuver has its unique application and result. For all of the maneuvers at less than 0.2 seconds BSTD, the no predictor case actually showed a small improvement over the predicted case. The differences can be contributed to the fact that these maneuvers are highly dynamic and for small BSTDs a predictor provides no benefit.

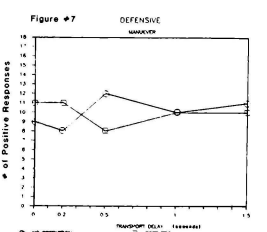
The high deflection gun shot (Figure 4) is a maneuver in which two aircraft are moving relative to each other at high relative speeds and rapidly changing aspect angles. Additionally, the weapon employment envelope is small for this case and the results correspond to the expectation that even with a predictor, no improvement is achieved. For this case, a delay of approximately 500 ms. appears to be acceptable. Figure 5 shows the results for the missile shot in which a simple IR-head is assumed to be locked on. The influence of BSTD of greater than 500 ms. is evident. The application of the first order predictor has little influence for this maneuver. Some points along this curve show the non-predicted data preferred for the small margin of BSTDs less than 500 ms. These differences are small and are most certainly the result of the fact that the maneuver is highly dynamic and evaluated subjectively. Figure 6, shows the results for the gun tracking shot. This is the maneuver in which the pilot is flying at a very high gain--very small corrections to position are required for a valid shot. As expected for this type of engagement, a small BSTD can be tolerated, approximately 250 ms. The benefits of applying the first order predictor are evident here, allowing the tolerable BSTD to increase to 500 ms.

The result for the defensive maneuver is shown in Figure 7. The results here point out one of the limits of this evaluation. We developed our BSTD algorithms based on Figure 3. However, IRIS is limited in that only a view from the attacker's aircraft is available in real time. With this as a limitation, we attempted to evaluate the BSTD effects on the defensive maneuver from the attacker's view point. As expected, an evaluation of the maximum BSTD allowable for the defensive maneuver from the attacker's viewpoint was inconclusive.









Figures 4-7. Pilot evaluations of BSTD effects on A/A engagements

Conclusion

The approach taken in conducting this study was to provide data that supports a common assumption, that the delays due to transmission times pose little, if any, degradation of training fidelity. American Telephone and Telegraph engineers estimate that data can be transmitted anywhere in the United States in approximately 20 ms. and 30 ms. to Europe. The primary source of an added delay may be that required to up/down link to a satellite. A satellite communications link provides greater coverage of networked locations but has delays approaching 250 ms. The results of this study show that, at least for the maneuvers tested, a delay of 250 ms. can be accepted with little degradation. Additionally, a simple first order predictor can extend this maximum delay to 500 ms. allowing additional delays for message encoding and interfacing circuitry. The defensive maneuver (DM) is the only maneuver tested that produced inconclusive results. However, the rate of change of aspect angles closely resembles that of the high deflection gun shot where 500 ms. of BSTD is tolerable. Additionally, the DM provides us with increased confidence in our delay algorithm derivations because the algorithms were developed for the attacker's viewpoint. The results of this study tend to support the conclusion that delays due to time for transmission may not affect network simulation training for the engagements investigated. The use of more sophisticated predictor schemes and data transmission techniques should provide realistic air-to-air two-ship relative positions with transport delays of up to 500 msecs.

PREDICTIVE COMPENSATION OF VISUAL SYSTEM TIME DELAYS

Donald J. Sobiski
IBM - Federal Systems Division
Owego, NY 13827

Frank M. Cardullo *
Watson School of Engineering
State University of NY
Binghamton, NY 13901

Abstract

Time delays in systems dependant on loop closure by a human operator in series with a visual display can cause degraded system performance and altered workload. A simulation of executive class jet aircraft dynamics with a flight director type display and programmable delays has been developed to explore these types of delays, and to develop and test a predictive method of compensation in an experimental environment. This predictive method uses the state transition matrix in a feedback loop to compensate for time delays. The experiment consisted of a compensatory tracking task which required nulling of the roll angle error of the simulated aircraft while it was being perturbed with random-appearing noise. A multi-dimensional measurement space was used to quantify performance through analysis of the experimentally obtained results. This analysis shows that the state predictor filter can achieve performance comparable to an undelayed, unfiltered system. A cluster analysis revealed that the delayed system with the state predictor filter and the original, undelayed system are fundamentally similar. A metric for evaluating workload is also presented.

Introduction

The capabilities of avionics systems are often enhanced by the addition of computers that can perform complex navigation and guidance tasks, and provide for higher levels of system integration. This performance enhancement carries a subtle penalty in the form of time delays added to system throughput. These delays can be lumped or distributed; the cumulative effect of uncompensated delays will be to degrade system performance. Tracking errors for humans occupied with high frequency tasks begin to increase with delays on the order of 100 ms.[20]

Compensation for the performance-reducing time delays caused by the systems that were designed to enhance effectiveness thus becomes a central issue in the design of modern avionics systems, and in the fidelity of related simulators. Indeed, these delays are a problem in any system, ground-based or airborne, with any man-in-the-loop digitally-based control system dependant on the display of system state.

Our research began with the question "can we make a system with time delays appear to be the same system without those delays?"

System Considerations

The loops closed by the pilot normally include visual display systems that convey aircraft state information, guidance information, flight director cues, and other mission related data. The pilot combines proprioceptive and vestibular inputs with display information to generate a control strategy. Temporal mismatches between these inputs can result in pilot induced oscillation (PIO), as in the pitch axis oscillation caused by processing delays in landings 5 and 7 of the space shuttle,[17] or can cause simulator fidelity problems.[9] A means of mitigating the effects of time delays is to employ compensation, which defines the loop structure in Figure 1.

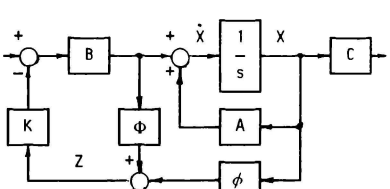


Figure 1. System Diagram

Temporal mismatch is brought about through time delays in processing. An example of a calculation of a time delay is as follows: the worst case time delay for a system running at 20 hertz (50 ms period), with a video display refreshed at 60 hertz (17 ms period) and pilot inputs being sampled each cycle is 134 ms (2 x 50 + 2 x 17). This occurs when the pilot input is made just after input sampling is executed by the system, so that the current input cannot be proc-

* Member AIAA

essed until the next computation cycle, and when the result of that delayed processing is sent to the video immediately after the video refresh has begun, forcing the data to be held until a new video cycle can access it. The minimum possible delay will be 67 ms, and assuming a uniform distribution on the input, the average system delays will be 100.5 ms with an uncertainty of ± 33.5 ms. The uncertainty in the time delay is a figure of merit not usually derived. If a subsystem performing a secondary process (i.e., location) is also running asynchronously at 20 Hz, the total worst case delay becomes 234 ms, with the growth in uncertainty related to the variance in the startup times of the system and the subsystem.

Frequency Domain Effects of Delay

Delays in the time domain and shifts in phase in the frequency domain are a Fourier transform pair. Frequency domain analysis provides a basis for examining the consequences of the effects of a time delay on a given dynamic system. Specifically, the relation:

$$f(t-t_d) \rightleftarrows e^{j\omega t_d} F(\omega) \qquad [1]$$

t_d = time delay
F = Fourier operator

allows calculation of the phase angle contribution of a time delay for frequencies of interest. Pilots "like" airplanes with integrator-like (k/s) closed loop dynamics and a frequency response (Bode' magnitude) that crosses the 0 dB axis between one and three radians/second, with a corresponding phase margin between 30 and 70 degrees.[1] They will provide the compensation (if possible) that is required to achieve this type of response.[14] For a nominal crossover of two radians/second, and using the above relationship, a time delay of 234 ms will add approximately 27 degrees of phase lag. This renders an undelayed system with 30 degrees of phase margin nearly unstable when delays are added. The pilot will generate additional lead to stabilize the system (if he can), but this increases his workload,[13] decreases his margin of safety, and raises Cooper-Harper ratings.[15] Compensation can add phase equalization, and can restore some measure of the desirable characteristics to the system.

Compensation

One approach to the compensation of time delays is the classical lead/lag filter design.[18] This strategy places a filter (compensator) of lead/lag type in series with the delayed output variable. Lead from the filter numerator cancels the phase lag added by the time delay, and the lag of the filter denominator smoothes noise amplified by the numerator terms. This type of design often requires a compromise be made between the amount of delay that is compensated, and amplification of the noise in the filtered output signal, and thus represents suboptimal compensation.

The approach described in this paper is to use state space techniques to design a predictor based on the transition matrix of the system. From the quadruple {A,B,C,D} that describes the linear, time-invariant, completely controllable and observable system in [2] with $U \in R^m$, $X \in R^n$, and $Y \in R^p$:

$$\dot{X}(t) = AX(t) + BU(t)$$
$$Y(t) = CX(t) + DU(t) \qquad [2]$$

The transition matrix Φ is:

$$\Phi(t,\tau) = e^{A(t-\tau)} \qquad [3]$$

The state, X(t), and the output, Y(t), can be completely determined for any time t, given initial conditions $X(t_0)$ and the control history $U(\tau)$, $\tau \in [\, t_0, t \,]$ by solving

$$X(t) = \Phi(t,t_0)X(t_0) + \int_{t_0}^{t} \Phi(t,\tau)\, BU(\tau)\, d\tau \qquad [4]$$

If the interval $[\, t_0, t \,]$ is chosen to equal a time delay t_d, then X(t) will be the value of the system at the end of the period of delay, and the result is a filter that functions as a predictor of the state variables in the system.

$$X(t) = X(t_0 + t_d) \qquad [5]$$

This is an heuristically appealing approach to use when time delays are present; it implies that the output of the state predictor filter will be temporally matched with the real time system outputs after time delays have been added in. When the future time history of the input is unknown, the convolution in [4] cannot be solved beforehand. However, reasonable assumptions about the form the input may take, i.e., piece-wise constant, sinusoidal, exponentially decaying, etc. do permit a priori closed-form solutions to be computed.[8] It will be demonstrated that using the state predictor filter with the assumption of piece-wise constant, linear inputs restores the phase and gain margin stability properties of the time-delayed system to those from the undelayed system, while avoiding noise contamination problems, and that delays as long as 800 ms can be adequately compensated.

Experimental Procedure

An experiment was developed after earlier work done by Ricard and Harris.[18] This work was a two axis

control task (pitch and roll) of a linear model of executive class jet aircraft dynamics with input disturbances, time delays of 200, 400, and 800 ms in the feedback loop, and lead/lag compensation on the output variables of aircraft attitude presented as an artificial horizon in a CRT display. The task consisted of controlling the attitude of the simulated aircraft using a two degree of freedom controller, while the aircraft model was perturbed by sinusoids of varying frequencies. The results from that experiment were used to develop "optimal" pole locations for filters that corresponded to different time delays, and also to infer phase lead requirements for pilots as a function of time delay.

Conspicuous by omission were lack of tests with no compensation and/or no delays; these tests would have served as a control, and provided calibration for the testing with filters.[4] The authors also did not validate (through testing) the filters with "optimal" pole locations that were the results of the experiment by testing those particular designs.

That experiment was simplified for this paper by restricting it to the lateral (roll) axis. This was to ensure isolation of cause and effect, and to eliminate the contribution of cross-coupling between axes to subject workload. The transfer function for the change in roll angle per deflection of control stick used in the experiment is (in Laplace notation):

$$\frac{\theta(s)}{\delta_e(s)} = \frac{314.04\,(s^2 + .8929s + 3.46)}{(s^4 + 7.1513s^3 + 9.163s^2 + 22.0625s)} \qquad [6]$$

This transfer function was used in the original experiment, and has been used by other researchers as well.[7]

Experiment Setup

The simulation was implemented in an 80286 based desk-top computer with an 80287 coprocessor also installed, resulting in parallel processing. The coprocessor did most mathematical operations while the 80286 CPU controlled I/O and updated the graphics display. An assembly language program managed protocol between the coprocessor (which runs an Assembler dialect) and a compiled BASIC program running in the CPU. The frequency of the simulation was approximately 35 Hz, and data was sampled once each period. The display was refreshed at 60 Hz. These rates are essential to avoid confounding experimental results by the addition of delays inherent in the simulation itself. Worst case system delays were (2 x 28 + 2 x 17) = 90 ms, with an average delay of 67.5 ± 22.5 ms. The subjects sat in front of a color graphics display and used a self-centering hand-held stick controller to generate aileron inputs to the simulation.

The display used in the experiment is represented in Figure 2. The horizon used to represent bank angle is the solid horizontal line in the center, while the lines to the left and right of it provided a reference for level flight. The horizon pivoted about its center point in response to the roll angle of the aircraft, which was filtered and delayed. The other elements of the display were animated during preliminary testing, however, they were made static during final testing to minimize distraction, to ensure that only one input-output relationship was being measured, and to reduce computational burden.

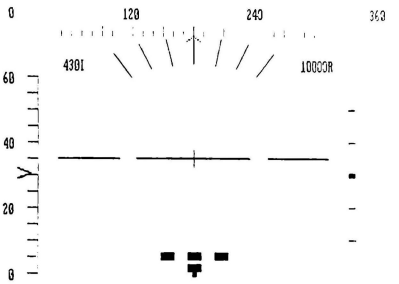


Figure 2. Display Representation

Noise Sources

The simulation was driven by zero-mean noise; several candidate noise sources were used in preliminary testing to evaluate the effects of frequency content and bandwidth on the performance metrics being developed. Driving the system with a noise input containing energy at frequencies above which a human subject can respond will cause errors in controlling the simulation while creating a sufficiently high workload to require full attention from the subject.[13] One noise input that was used was a sum of thirteen sinusoids; this has been offered as random appearing noise source whose power spectrum is approximately rectangular, and has a 2 radians/second cutoff frequency.[11] Two radians/second has been shown to be near the upper frequency that human subjects can completely follow.[13] This noise source has been developed in the continuous time domain, however, and does not retain the appropriate power spectrum characteristics when it is implemented digitally, as Figure 3 illustrates. Specifically, the dominant frequency content extends to nearly five radians/second, and energy appears out at 39 radians/second. This noise source was used in testing, however, because of its familiarity to many researchers.

The characteristics of the 13 sines noise source motivated developing a method of producing "custom-

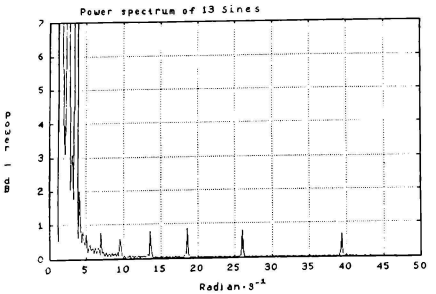


Figure 3A. Power Spectrum - 13 Sines

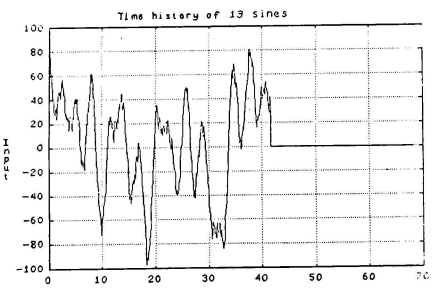


Figure 3B. Time History - 13 Sines

ized" noise sources directly in the digital frequency domain. These noise sources were designed by specifying the shape of the envelope of the desired frequency response, and then sampling within that envelope at a frequency equal to the period of the simulation.[19] A random number generator was used to do the sampling. The specific sampling frequency is:

$$\Omega_s = \frac{2\pi}{N\ T_s} \qquad [7]$$

N = the length of the IFFT
T_s = the period of the simulation

The resulting frequency samples were zero-padded to produce a vector of length 1024, concatenated with a mirror image of this new vector to create a strictly real time function, and finally Inverse Fast Fourier Transformed (IFFT) to produce a discrete time history whose power spectrum has the proper characteristics. Figure 4 shows the sampled frequencies within a design envelope described by an exponential with a half power point at a frequency of four radians/second, and a cutoff frequency at eight radians/second. The time history of this frequency domain designed noise signal obtained from an IFFT is shown as Figure 4B. The power spectrum of the time signal is presented in Figure 4C to illustrate the preservation of the original exponential envelope and the absence of any frequency content outside the specified region. The absence of the very high frequency jitter (present in the 13 sines noise) reduced the subject's perception of workload, while still providing the stimulus for full attention. This frequency domain design approach to generating input sequences for testing offers much greater control of the characteristics of the driving noise, and deserves further study.

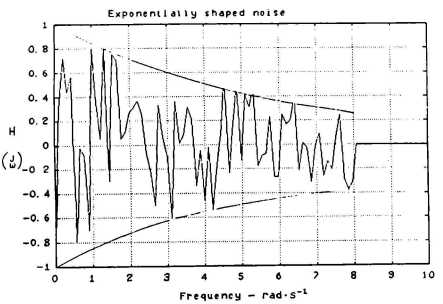


Figure 4A. Design - 8 Rad Noise

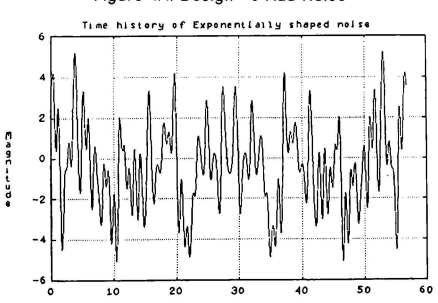


Figure 4B. Time History - 8 Rad Noise

Experimental Procedure

Fifteen subjects participated in preliminary testing, used to validate procedures, develop metrics, examine noise sources, and refine techniques. Two of these subjects were pilots, and a third had piloting experience. Nine subjects participated in the final testing, of which one was a pilot. Only the results

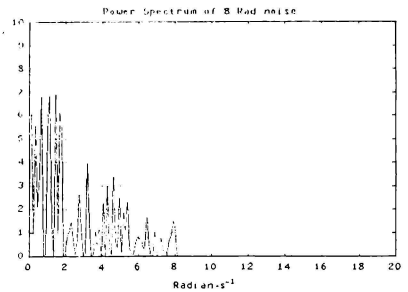


Figure 4C. Power Spectrum-8 Rad Noise

from final testing will be discussed in this paper. All of the subjects in the second group had simulator experience, albeit limited in most cases. Training was accomplished under subject control; each subject was allowed to set up the different test variables and practice with the simulation until familiarity and confidence were obtained. Training to asymptote was not undertaken; each individual's subjective measure of readiness was used to determine if sufficient training had been accomplished.

Test types Two groups of seven trials were completed by each subject. Each trial lasted two minutes, and was initiated by the subject. Four sets of tests were established, with three tests in each set. The four sets of experiments were:

1) with no filter on the output

2) with the lead/lag filter, defined by the Ricard and Harris experiment, on the output

3) with a full state feedback predictor filter with state estimation (presented below)

4) with a reduced order predictor filter on the output (presented below).

The three tests within each experiment were for the three delays of 200, 400, and 800 ms. The software-controlled delay was inserted in the feedback loop between the output of the model, or filter (Figure 1) and the change in the aircraft bank angle indicated by the position of the horizon in the display. These twelve tests were driven by the 13 sines noise sequence. Two additional tests were added as controls - these two tests had no delay or filter; one was driven by the 13 sines noise and the second was driven by the 8 radians/second exponentially shaped noise. The bank angle of the aircraft and the position of the stick were recorded at 35 Hz for all tests for use in subsequent data analysis. The order in which the tests were administered was established with a random number generator, and this ordering was held constant for all subjects.

Predictor Design

A block diagram of the system with the predictor (Figure 5) proceeds from an implementation of equation [4], with the matrix A representing a concatenation of the subject, the aircraft model, and the time delay. The time delays in the subject and in the system were modelled as linear approximations by substitution of the second order Pade' approximation:

$$e^{X} = \frac{(X^2 + 6X + 12)}{(X^2 - 6X + 12)} \qquad [8]$$

The transfer function of the subject was chosen to match a lateral control task performed with a rate controller,[21] and included a term representing the lumped time delays of the subject:

$$H_p = \frac{18\,(s+1)\,e^{-.3s}}{(s+3)(s+9)} \qquad [9]$$

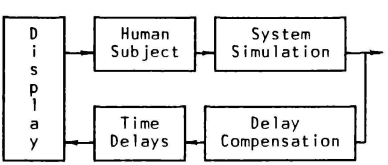


Figure 5. Predictor Structure

The matrix ϕ is the state transition matrix, and the matrix Φ represents the convolution of ϕ and the input. It is this convolution that provides the high degree of noise immunity in the state predictor design. The non-standard feedback loop closed around the input through Φ is presented as a significant element of the predictor filter. K is an (m by n) matrix that provides for a conformal loop, and Z is the predicted state vector. When K is chosen to equal C, the output matrix, the predicted output will be the feedback term. When system delays are large, a better choice for K can be made to provide for pole placement, specifying that the closed loop poles be made equal to the closed loop poles of the original, undelayed system. This may be a better choice because the low bandwidth of the linearized time delay models introduces errors into the prediction equations. This

modelling error is shown in Figure 6, which represents the phase angle from the Pade' approximation [8] plotted with the phase angle of e^{jwt_d}, for t_d = 800 ms, and illustrates the divergence of the approximation at frequencies above 3 radians/second.

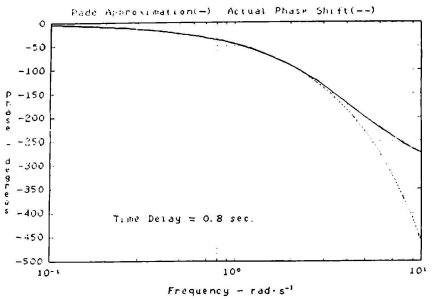


Figure 6. Pade' approximation and Arg [e^{jwt}]

Two versions of the state predictor filter were designed and tested - one version used full state feedback to develop the control laws, and resulted in an eleventh order filter design (four aircraft states, four pilot states, three time delay states). Since the states of the pilot and of the time delay (included as part of system model for predictor design) were not available, state estimators of these seven states were implemented as part of the full state predictor feedback design. The second predictor filter was designed using output feedback, however, the system transfer function is a non-minimum phase transfer function and a stable, full order filter that reconstructs the system states is not possible to design. An infinite number of reduced order filters that approximate the full order output filter exist; a third order filter was chosen and implemented. This reduced order filter does not provide for all the compensation the full state predictor offers, however, it was included because it offers reasonable compensation and is extremely simple to implement.

Design Analysis

The implementation of the three filter designs was preceded by frequency analysis and step responses. The Bode' plots that follow preview the performance that may be anticipated from each design. The performance obtained was slightly different because the subjects do not behave exactly as the pilot model predicts, and because the lead and gain equalization generated by the subjects in response to the time delays is not reflected by adaptation in the pilot model.

Undelayed System with Pilot

Figures 7A,B shows the Bode' plots of the original, undelayed system of the aircraft and pilot models in series. These plots define the desired response characteristics for the design of the compensation for the delayed system. Note the desired integrator-like (k/s) crossover behavior of the magnitude and the corresponding phase margin of 55 degrees.[14] This indicates that the system has the proper characteristics for stable flight and good performance.[1]

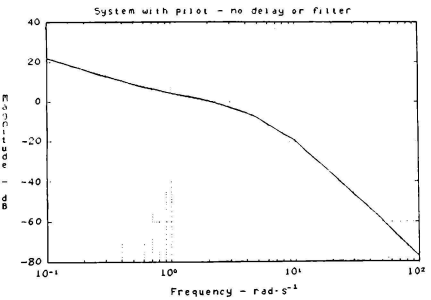


Figure 7A. Magnitude - no delays

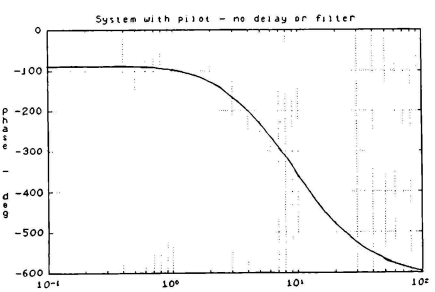


Figure 7B. Phase - no delays

When time delays are added to the original system, the magnitude plot remains the same, but phase is increased by the factor of ωt_d radians. This equates to 23 degrees of lag at 200 ms delay, 46 degrees of lag at 400 ms delay (at which point the system is marginally stable), and 92 degrees of lag at 800 ms delay, which destabilizes the system.

400 ms Delay, Lead/Lag Filter

The lead/lag filters are of the form:

$$\frac{(\tau_n s + 1)}{(\tau_d s + 1)} \qquad [10]$$

The following table presents the zeros, poles, maximum phase lead (degrees), and frequency (radians/second) at which the maximum phase lead occurs for the three "optimal" lead/lag filters used.[18] Note that the zero of each filter equals the inverse of the time delay for which it was designed.

zero	pole	lead	freq
5.	5.38	2	5.2
2.5	4.75	18	3.5
1.25	5.90	40	2.5

Table 1. Lead/Lag Constants

Frequency Analysis The magnitude curve in Figure 8A shows the 0 dB crossover has risen to 3 radians/second, the upper limit of pilot acceptance. The filter transfer function does not contain a term to adjust for steady state gain; the gain distortion is the penalty for additional phase lead.[7] The magnitude plot also shows a shift in the -40 dB per decade "knee" of the response curve to a point near the 0 dB crossover frequency, and thus transition from k/s -like behavior to k/s^2 -like behavior is also occurring near crossover. This will lower the effective damping ratio in the closed loop system by creating a pair of complex poles at the crossover frequency.[5] The lowering of the damping ratio is also manifested in the phase plot by the reduction in phase margin at the crossover frequency.[16] The phase plot also shows this system is clearly unstable at the crossover frequency of 3 radians/second, which will force the pilot to lower his gain and/or supply additional lead to stabilize the system.

This analysis indicates that subjects will probably be less successful in controlling this system than in controlling the undelayed system. Furthermore, the phase margin of this design at 2 radians/second is roughly the same as the original system at this same delay, which indicates no real gain in performance by the addition of compensation has been accomplished. Indeed, because of the additional gain equalization required, and the lower system damping ratio, the system with the lead/lag filter should prove harder to control than the unfiltered system at this delay! This was observed and commented on by the subjects, and also reflected in the results obtained. These trends were observed in the designs for all three lead/lag filters, and indicates that *ad hoc* design techniques should be approached with caution.

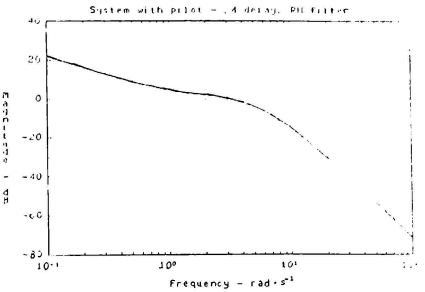


Figure 8A. .4 Delay - LD/LG filter

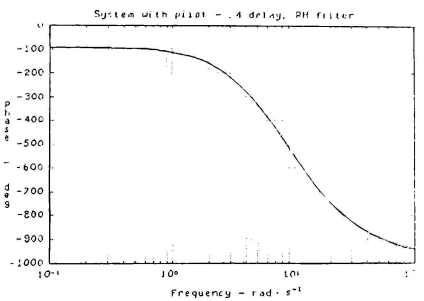


Figure 8B. .4 Delay - LD/LG filter

400 ms Delay, State Predictor Filter

The state predictor designs assumed a piece-wise constant input by the subject, as if a first order hold of duration t_d were placed on the input. This is a reasonable approximation for the time delays used in the experiment, given the bandwidth of human response.[13] This assumption permitted the prediction time to be set equal to the time delay, and a closed form solution for Φ to be calculated for t_d of 200, 400, and 800 ms.

$$\Phi = \int_0^{t_d} \phi(\, t,\tau\,)\, d\tau \qquad [11]$$

When this filter design was inserted into the feedback loop, an analysis (not presented) showed that the Pade' approximation for time delay introduced errors at higher frequencies (Figure 6) for the delays of 400 and 800 ms, and the phase angle characteristics of the original system were not completely replicated. The gain matrix K was then calculated to place the

poles of the delayed system/state predictor at the original undelayed system pole locations.[2] This resulted in the desired phase and gain properties in the filter design, and overcame the limitations of the linearized time delay.

The implementation of the full state feedback predictor design is a matrix equation represented by Figure 5:

$$\text{with} \quad G^{-1} \triangleq (I_m + K \Phi B) \qquad [12]$$

$$\dot{X}(t) = (A - B G K \phi) X(t) + B G U(t) \qquad [13]$$

The implementation of the reduced order filter was a third order transfer function with the numerator and denominator approximately equal to the transfer function formed by the last four terms of K as numerator coefficients for descending powers of s and the last four terms of C as denominator coefficients for descending powers of s.

$$H(s)_{reduced} = \frac{\prod_{i=1}^{3}(s - z_i)}{\prod_{j=1}^{3}(s - p_j)} \qquad [14]$$

Frequency Analysis Figures 9A,B shows the frequency response of the full state predictor design. The magnitude plot retains the 2 radians/second crossover frequency, and also the desired integrator-like (k/s) crossover characteristic. An astonishing result is that the -40 dB per decade knee is completely gone from the response over this range of frequencies; in fact, the system response almost totally resembles a pure integrator! The phase angle response of the undelayed system has been recovered by the state predictor, and the phase margin of 55 degrees has been completely restored. This result, in concert with the change in the magnitude response plot, indicates that the damping ratio of the system has either remained the same, or increased, in spite of the 400 ms time delay. This implies that the delayed system with the state predictor filter should provide performance similar to the original system. This behavior was observed and commented on by the subjects, and also is evident in the data.

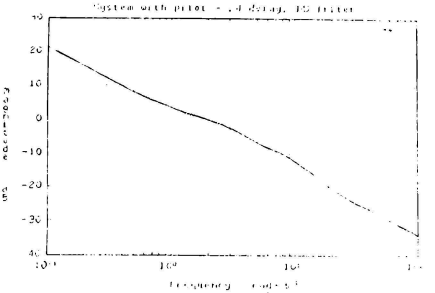


Figure 9A. .4 Delay - State Predictor

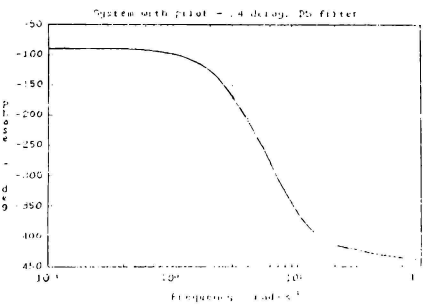


Figure 9B. .4 Delay - State Predictor

Results

System dynamics and filter transfer functions were discretized with the impulse invariant transform method of mapping the s-plane to the z-plane, which retains the proper response dynamics in a digitally implemented system. Measurements of the position of the control stick and of the displayed roll angle were made once each cycle, and the following quantities were calculated from those measurements for each subject:

1) Mean of roll error
2) Standard deviation of roll error
3) Correlation between stick and roll

Power Spectrum quantities

4) Total remnant energy
5) Center frequency of remnant
6) Standard deviation of remnant
7) Center frequency of stick
8) Standard deviation of stick
9) Center frequency of roll error
10) Standard deviation of roll error

This resulted in 9 data sets (one per subject) containing 14 vectors (one per test) of 10 dimensions each. A tenth data set was created by calculating the mean of these 10 metrics across the results of all 9

subjects; the associated standard deviations were also calculated for statistical significance.

No attempts were made to optimize the predictor designs or to improve performance by "tweaking" gains or making other adjustments; the filters were implemented exactly as the design theory dictated.

Performance

The mean roll error was rejected as a measure of performance for this type of testing - since the input sequence was zero-mean noise, the mean roll error was also near zero. The mean roll error would be near zero even if a (hypothetical) system was in PIO with a bank angle amplitude of ± 30 degrees, which illustrates the uselessness of mean error as a measure of performance in zero mean systems. The total range of mean roll error across all filters and all delays was only ± 1.5 degrees.

The performance criterion that was chosen was the measure of the range of error that represented the subject's ability to control about the mean, or wings level attitude. This measure is the standard deviation of roll error. Good control (and hence, good designs) should allow tight control about the wings level attitude when disturbances are present, and poor control (and poor designs) should be hallmarked by large variations in attitude about the wings level attitude.

Figure 10 is a plot of the standard deviation of each filter at each time delay, and includes a plot of the no-filter results, as well. The abscissa of the plot is the time delay in seconds, and the ordinate is the standard deviation in degrees.

The no-filter plot illustrates expected behavior; the standard deviation of error grows as the delays become larger, which clearly indicates that the system is becoming more difficult to control. At 200 ms of delay, the lead/lag design and the reduced order predictor offer no clear advantage over not filtering at all; this result is caused, in part, by the ability of the subjects to provide phase and gain equalization. The full state predictor shows a 23% reduction in deviation - this improvement is due to adequate compensation by the design, and perhaps to the increased damping ratio that results from its implementation.

At 400 ms of delay, both predictors provide greatly reduced deviations, which translates directly to improved performance. The reduced order predictor shows nearly 40% reduction in error, however, the full state predictor should be able to produce similar results with some design optimization. The lead/lag filter does not offer any improvement over not filtering at all.

The lead/lag filter becomes nearly uncontrollable at 800 ms delay; this possibility was noted in Figure 8 for the 400 ms design, and those same trends are present here. The predictor designs continue to demonstrate more than 30% reduction in deviation, and the full state predictor shows performance that is very comparable to the original, undelayed and unfiltered system.

Filter	None	Ldlag	Rduced	Full
symbol	0	X	+	*
delay				
0	13.5	x	x	x
200	15.6	17.1	15.9	12.1
400	17.7	17.7	10.7	12.9
800	23.0	28.1	16.6	15.1

Table 2. Standard Deviation of Roll Error

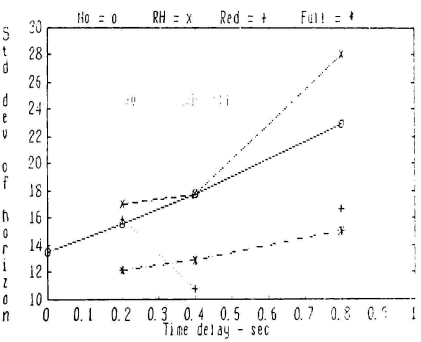


Figure 10. Standard Deviation of Roll Error

Cluster Analysis

The ten metrics calculated from the data were used to define a ten dimension measurement space, with the data from each subject defining a point in the space that corresponded to each test. The overall means of each test were also placed in the space, making a total of 140 data points in the space. Cluster analysis attempts to find patterns in the data based on some rule. In Forgy's algorithm[10] the rule is the 2-norm (Euclidian norm), and it is applied as follows:

Forgy's Algorithm 1) Starting with an arbitrary number of clusters, assign data points arbitrarily to clusters in the space until all points have been assigned.

2) Calculate the centroid of each cluster i by finding the mean in each dimension k:

$$\text{Centroid}(\, i,k\,) = \frac{1}{N_i} \sum_{n=1}^{N_i} P_i(n,\, k) \qquad [15]$$

P_i = all points in Cluster i
N_i = the number of points in the cluster i

3) Each centroid now represents the center of a new cluster; reassign each point P in the space to the cluster it is closest to by using the rule:

$$\text{Cluster}(\, m\,) = \text{all } P_i \ni ||Centroid_m - P_i||_2 < ||Centroid_k - P_i||_2 \text{ for } k \neq m \qquad [16]$$

4) Go to step 2) and repeat the process until no points change clusters in step 3).

Forgy's algorithm must converge in a finite number of iterations for a finite number of points - at each iteration, only those points move that decrease the total error in the space; **no** points move that will cause the error to increase. Because the set is finite, an iteration will be reached where no point can move without increasing the total error.

The algorithm is made adaptive in the following manner: If any cluster is detected at step 4) that contains only the centroid, then that cluster is deleted, the total number of centroids is reduced, and then the procedure continues. Conversely, if a "large" cluster is detected, it can be split, two new centroids are calculated, and then the procedure can continue. These changes help to find "natural" groupings in the data.

The result of a clustering algorithm is a data set that is arranged in clusters which have the property that every point in a cluster is closer to that cluster's centroid than to any other centroid in the space. In other words, all the members of a cluster are more similar to each other and less similar to the members of any other cluster in the space, as determined by the sorting rule and the characteristics of the space.

Cluster analysis was performed for arrangements of 3, 4, 6, 8, 9, 11, and 13 clusters in the space. A sample of these results is presented as Table 3. This set of results was obtained by assuming 6 clusters are in the space. In processing, one cluster was deleted, and the contents of the remaining clusters are shown. Interpretation of the table is illustrated by the following example: Cluster one contains 15 members; 4 members are the results of the unfiltered testing - 2 from the 400 ms delay tests, and 2 from the 800 ms tests, ten members are from the lead/lag testing at 800 ms of delay, and 1 member is from the reduced order predictor with 200 ms of delay. These results in this cluster all resemble each other in terms of the metrics defining the dimensions of the space.

	Filter	None	Ldlag	Rduced	Full
Cluster	delay	number of members			
ONE	0	0	x	x	x
	200	0	0	1	0
	400	2	0	0	0
	800	2	10	0	0
TWO	0	3	x	x	x
	200	1	2	1	0
	400	5	7	0	1
	800	1	0	2	1
THREE	0	0	x	x	x
	200	2	4	2	0
	400	3	3	0	0
	800	7	0	2	0
FOUR	0	5	x	x	x
	200	3	4	4	2
	400	0	0	1	5
	800	0	0	5	7
FIVE	0	2	x	x	x
	200	4	0	2	8
	400	0	0	9	4
	800	0	0	1	2

Table 3. Cluster membership

For all the cluster arrangements specified, **nearly all the results from the two predictor filters were consistently grouped with the unfiltered results with 0 and 200 ms delay,** as in clusters 4 and 5, and the lead/lag filter results were consistently grouped with the unfiltered system results from the 400 and 800 ms time delay experiments, as in clusters 1,2, and 3. **The implication is that nearly all the results from the predictor filters "look like" the unfiltered results from little or no delay.** This indicates that the answer to the original question, "can we make a system with time delays appear to be the same system without those delays?" is YES!

Workload Metric

This was a high workload task, using high frequency noise which ensured perfect signal following could not be accomplished by human subjects. As time delays increased in the system, workload also increased. This workload consisted of the objective

workload, and the subjective workload. The objective workload consists of the physical aspects of performing a task - how much energy is needed, how much attention is required, etc. A high workload task may not be difficult to perform. The subjective workload is a perception of how well the subject is doing, and is related to the stress that the task induces. If two tasks have the same objective workload, but a subject can perform well at the first task and fails at the second, the second task will be perceived as having a greater total workload.

The calculations of the center frequencies for the driving noise, stick and roll error provide a relationship that helps quantify these two terms. The center frequencies are those frequencies at which the mean of the power spectral density occurs. The specific relationship observed in this experiment was:

$$\omega_{stick} > \omega_{noise} > \omega_{error} \qquad [17]$$

Where ω is the center frequency of the variable subscript. This relationship can quantify workload in the following manner: as those inequalities become equalities, objective workload increases. If any inequalities reverse, subjective workload increases, and performance will degrade.

The following ratios were obtained from Figures 11A,B, and Table 4, and are offered as preliminary values for workload measurement. Adequate performance can be attained at reasonable workload levels when the ratio between ω_{stick} and ω_{noise} is greater than 1.50, and the ratio between ω_{noise} and ω_{error} is greater than 1.25. Heuristically, a well-behaved system will have the following properties: the system will attenuate high frequency noise, thus, ω_{error} will normally be lower than ω_{noise}. The center frequency for the stick, ω_{stick}, must be higher than the noise for the following reason: if a sine wave is used to track a sine wave of the same frequency, and any error ever occurs, that error can never be nulled. A higher frequency (faster) tracking sine wave is required to "catch up" with the original reference signal.

The curves for both predictor filters and for the unfiltered system in Figures 11A,B show that the center frequency of the stick decreases as time delays get larger. This is a consequence of the time delay - the subject must apply the control for longer period of time before a reaction is apparent. This has been observed before, but apparently not well understood.[8,11] The center frequency of the error, ω_{error}, increases with delay, and begins to approach the center frequency of the noise, which was constant at 0.9861 radians/second. This can be accounted for by the unnecessarily large inputs that are occurring because the control is applied for longer and longer

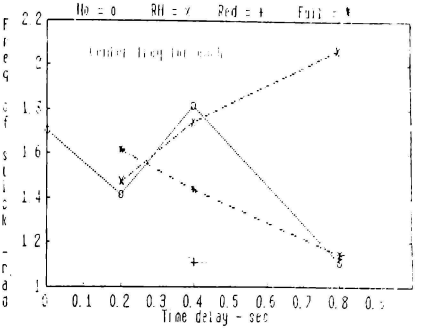


Figure 11A. Input Center Frequency

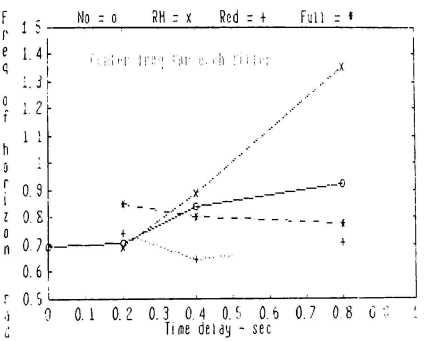


Figure 11B. Roll Error Center Frequency

periods. In general, the triplet inequality proposed above is approaching unity as the time delays increase; subjects generally observed that the work load for the predictor filters was increasing, but was not onerous because they were able to do very well with these designs. The curve of the unfiltered system in Figure 11B, and the data in Table 4 shows that at 800 ms of delay, the triplet inequality is nearly an equality - most subjects commented that they had difficulty controlling this test, and thought it was one of the more difficult control tasks. The plot for the lead/lag filter shows the results of the system becoming unstable; when the delays become longer, the subject actually excites the system with input frequencies above that of the noise - that these frequencies come from the subject is evident in the curve for the stick input for the lead/lag filter in Figure 11A. The lead/lag curves show a reversal in the triplet inequality occurs at 800 ms of delay - this test was impossible to control, and subjects realized this almost immediately.

Filter	None	Ldlag	Rduced	Full
delay	Stick to Noise ratio			
0	1.73			
200	1.43	1.49	1.49	1.64
400	1.83	1.77	1.12	1.46
800	1.12	2.10	1.15	1.16
	Noise to Error ratio			
0	1.42			
200	1.40	1.42	1.33	1.16
400	1.83	1.11	1.53	1.22
800	1.07	0.73	1.40	1.26

Table 4. Triplet Inequality Ratios

Conclusion

A predictive filter for compensating for system time delays has been presented. This filter uses the state transition matrix in a feedback loop to compensate for delays. Frequency analysis shows the state predictor filter can restore phase and gain margins to the system for delays as long as 800 ms. Two different predictor designs were implemented in an experiment that extended earlier work - one design used full state feedback and state estimation, and the second used output feedback in a reduced order design.

A multi-dimensional measurement space was defined and used to quantify performance through analysis of the experimentally obtained results. This analysis shows that the state predictor filter can produce performance akin to an undelayed, unfiltered system for delays as long as 800 ms. A cluster analysis using an adaptive version of Forgy's algorithm revealed that the delayed system with the state predictor filter and the original, undelayed system are fundamentally similar. A metric for evaluating workload in terms of the triplet ratio between the center frequencies of input, noise, and system response was presented.

References

1) Ashkenas, I.L., Twenty Five Years of Handling Qualities Research, J. Aircraft Vol. 21 No. 5 May 1984

2) Brogan, W.L., Modern Control Theory, Quantum Publishers, Inc. New York, NY 1974

3) Cadzow, J. A., Discrete Time Systems, Prentice-Hall Englewood Cliffs, NJ 1973

4) Campbell, D.T., Stanley, J.C., Experimental and Quasi-experimental Designs for Research, Rand McNally College Publishing Co. Chicago 1983

5) Canfield, E.B., Electromechanical Control Systems and Devices, Kriegler Publishing Co. New York 1977

6) Cooper,F.R., Harris, W.T., and Sharkey, V.J., Effects of Visual System Time Delay on Pilot Performance, Proceedings of NTEC Conference Orlando 1975

7) Crane, D.F., Flight Simulator Visual-Display Delay Compensation, 1981 IEEE Winter Simulation Conference Proceedings

8) Grunwald, A.J., Predictor Laws for Pictorial Flight Displays, J. Guidance Vol. 8 No. 5 Sep-Oct 1985

9) Gum, D.R., and Albery, W.B., Time-Delay Problems Encountered in Integrating the Advanced Simulator for Undergraduate Pilot Training, J. Aircraft Vol. 14 No. 5 Apr 1977

10) Hartigan, J.A., Clustering Algorithms, Wiley New York 1975

11) Hess, R.A., Effects of Time Delays on Systems Subject to Manual Control, J. Guidance Vol. 7 No. 4 July-August 1984

12) Manes, S., The Oven of the Half Baked Idea, PC Magazine Vol. 6, No. 4, Febuary 24, 1987

13) Poulton, E.C., Tracking Skills and Manual Control, Academic Press, NY,NY 1974

14) McRuer, D.T., Graham,D., Krendel, E., and Reisener, W., Human Pilot Dynamics in Compensatory Systems, Air Force Flight Dynamics Laboratory Technical Report no. AFFDL-TR-65-15 July 1965

15) McRuer, D.T., Development of Pilot-in-the-Loop Analysis, J. Aircraft Vol. 10 No. 9 Sep 1973

16) McRuer, D.T., Krendel, E.S., Mathematical Models of Human Pilot Behavior, AGARD AG-188 Jan 1974

17) McRuer, D.T., Johnson, D.E., and Myers, T.T., Space Shuttle Flying Qualities Criteria Assessment, Phase IV - Data Acquisition and Analysis, Systems Technology, Inc. TR-1206-1 Nov. 1984

18) Ricard, G. L., Harris, W.T., Time Delays in Flight Simulators: Behavioral and Engineering Analyses, J. Aircraft Vol. 17 No. 3 Mar 1980

19) Schwarz, M., Shaw, L., Signal Processing: Discrete Spectral Analysis, Detection, and Estimation, McGraw-Hill 1975

20) Sevier, J.A., Minturn, D.B, Bernard, D.W., amd Pollard, T.J., The Effect of Computational Time-Delays on Pilot Performance In Real-Time Flight Simulation, Proceedings of the AIAA 22nd Aerospace Sciences Meeting Reno, NE 1984

21) Sobiski, D.J. and Male, H.W., Piloted Simulation Evaluation of the Combat Talon II Flight Director and Guidance Functions, IBM Corp., October 1986

FREQUENCY RESPONSE IDENTIFICATION OF A COMPUTER-GENERATED IMAGE VISUAL SIMULATOR WITH AND WITHOUT A DELAY COMPENSATION SCHEME

87-2425

Wayne F. Jewell, Principal Specialist, Member AIAA
Warren F. Clement, Principal Research Engineer, Member AIAA
Jeffery R. Hogue, Principal Specialist, Member AIAA

Systems Technology, Inc.
2672 Bayshore Parkway, Suite 505
Mountain View, CA 94043

ABSTRACT

Most modern flight simulators employ computer-generated images (CGIs) to display outside visual cues to the pilot. While these CGIs have excellent low-frequency characteristics, their high-frequency bandwidth is inherently limited by the update rate of the combined host computer and visual simulator. In addition, the complex architecture of the overall system (i.e., combined simulator/CGI system) makes the analysis of the frequency response extremely difficult. The frequency response technique presented in this paper permits the dynamics of the overall system to be measured and analyzed very easily. The technique, which is based on hardware and software that are compatible with an IBM PC, identifies the simulator response from the pilot's control inputs to the output of the CGI, as measured by a photometric sensor attached to the visual display monitor. The data show that a novel delay compensation scheme can extend the bandwidth of CGI visual simulators. The frequency response identification technique could be used to document further improvements to visual simulators such as advanced architectures and parallel processors.

INTRODUCTION

Several attributes of computer-generated images (CGIs) have been identified[1-7] as affecting pilots' judgments of the fidelity of external visual field simulation for precision flight control tasks. Among the attributes thus identified are: image delay, image content, field of view, and image perspective. Particular attention has recently been focussed[1] on compensation for the image delay, which may otherwise seriously compromise the achievable closed-loop bandwidth of piloted control in precision hovering and stationkeeping tasks with simulated aircraft.

Measurements of the state transmission dynamics (from vehicle state change to visual image motion change) have been made on the CGI visual scene simulator of the National Aeronautics and Space Administration (NASA) Ames Research Center (ARC) Vertical Motion Simulator (VMS) in the context of two piloting tasks, viz., pitch attitude and heading control, both of which are essential in hovering and stationkeeping of rotorcraft. The measurements were made both with and without a novel delay compensation scheme that was designed to increase the bandwidth of the visual simulation. The details of the delay compensation scheme are thoroughly discussed[8] and will not be repeated herein. It is sufficient to say that the compensation scheme is designed to eliminate the phase lag due to time delay in the CGI with virtually no magnitude distortion up to frequencies of about 15.0 rad/sec, the limit of measurement in these tests with excitation by a human pilot. It is thought that this is sufficient visual state bandwidth for the purpose of real-time, man-in-the-loop simulation for flying qualities research.

The term "visual state bandwidth," as used here, refers to the frequency bandwidth characterizing changes in the rate of movement, the orientation, or the position of a visual image, e.g., the horizon, from which a state of the vehicle's orientation can be perceived by the human operator. "Visual state bandwidth" is not to be confused with the (usually much higher) spatial frequency bandwidth of the video modulation transfer function describing the visual resolution and luminance contrast of image details and texture provided by electro-optical visual systems[9]. If the perception of a state of the vehicle depends critically on visual image details (which are governed by the spatial frequency bandwidth of the video modulation transfer function), the "visual state bandwidth" may be limited by the "spatial frequency bandwidth." Some of the pilot opinion ratings from flight tests[7] imply that the pilots were exposed to test conditions wherein the visual spatial frequency bandwidth of outside cues did, in fact, limit the visual state bandwidth required for low-speed and hovering operations with a rotorcraft.

The major components of the VMS and the data measurement system are shown in the functional block diagram of Fig. 1. Under Contract NAS2-12414 with the Aeroflightdynamics Directorate of the U.S. Army Aviation Systems Command at ARC, Systems Technology, Inc., (STI) was asked to compute frequency responses of the simulated UH-60A and to compare them with frequency responses computed from flight test data. For the ground-based simulation, this included frequency responses of the UH-60A mathematical model (from δ_s to x_a in Fig. 1) at various airspeeds as well as frequency responses of the visual and motion simulators used on the VMS. This paper will restict itself to the results obtained while identifying the frequency response of the visual simulation. Other results of this contract will be reported separately by STI and the U.S. Army.

The method used to identify the frequency response of both the actual and the simulated UH-60A was to have the pilot generate "frequency sweeps" of each of the four controls, one at a time, while stabilizing the vehicle with the other three controls. This technique has been used successfully by STI and others to identify the dynamic response of many flight vehicles[10,11]. The vector of four controls is defined as

$$\vec{\delta}_p = (B_p, A_p, P_p, C_p)^t$$

where B_p is the longitudinal cyclic, A_p is the lateral cyclic, P_p represents the pedals, and C_p is the collective control displacement.

The subscript "p" in the above definitions stands for controller deflection at the pilot's station. In the simulator, these controller deflections are digitized and used as inputs to the mathematical model of the UH-60A, as shown in Fig. 1. The digitized controller deflections and aircraft states are given the subscript "d." In order to interface with some components of the simulation, the digitized aircraft states are converted into analog signals (e.g., the motion system and cockpit instruments). We will use the subscript "a" to represent the analog aircraft states and controller deflections.

The block digram of Fig. 1 shows how the digital and analog signals are used to drive the inputs to the visual and motion simulators. The output of the CGI (denoted by the subscript "v") is quantified by the STI visual display sensor (discussed below), and the output of the VMS (denoted by the subscript "m") is quantified by rate gyros and accelerometers. The aircraft states, controller positions, CGI, and VMS outputs are then recorded on a personal computer (PC), as shown in Fig. 1. The PC has a fast Fourier transform (FFT) program that permits the describing function between any two points in Fig. 1 to be computed. Various items of graphics software on the PC also permit the "raw" data as well as the processed results to be examined immediately following the collection of the data.

COMPUTATION OF THE NET THROUGHPUT DELAY

In order to check out the various items of hardware and software associated with the components shown in Fig. 1, a simple test to compute the effective throughput delay of the combined ADC-CDC 7600-DAC system was devised (from point δ_a to δ_{acm} in Fig. 1). From Fig. 1, the transfer function to be identified is given by

$$G_a = G_{adc} * G_{cdc}(1) * G_{dac}$$

where $G_{cdc}(1)$ represents the throughput delay of the CDC 7600. The notation $G_{cdc}(x/\delta)$ will be used to represent the transfer function from the control δ to the state x.

The HP function generator was used to obtain a forcing function for δ_{acm}. The signal δ_{acm} was input to the ADCs at points 1 and 3 shown in Fig. 1, and the describing function between δ_a and δ_{acm} was then computed using the FFT software on the PC. The resulting frequency response for a CDC 7600 sample period of 28 ms is shown in Fig. 2. Only data points with a coherence of 0.70 or higher are included in Fig. 2. The recording frequency used by the PC for these tests was 50 Hz (the Nyquist frequency is 157 rad/sec). Note that the response shown in Fig. 2 is flat out to 50 rad/sec, with some gain distortion at 60 rad/sec that is probably due to sampling effects.

Note from Fig. 2 that the effective throughput delay from δ_{acm} to δ_a can be approximated by a pure time delay of 55 ms when the CDC sample period is 28 ms. The test was repeated for a CDC 7600 sample period of 50 ms. The results for both tests are summarized in the plot shown in Fig. 3. The two data points at 28 and 50 ms represent repeated runs at the same frame time for the CDC 7600 sample period, and the straight line fit is an average of the data points.

FREQUENCY RESPONSE OF THE PHOTOMETRIC SENSOR

The frequency response of the photometric sensor was measured separately from the tests conducted on the VMS. These tests were conducted by providing known inputs to the photometric sensor and measuring the outputs of the photometric sensor on a carthode ray tube (CRT). The test scenarios for the pitch and yaw axes are shown in Fig 4. The resulting frequency response of the photometric sensor are summarized in Fig. 5. The frequency response of all measured visual states must be corrected by the magnitude and phase angle shown in Fig. 4.

FREQUENCY RESPONSE OF THE MATHEMATICAL MODEL

The frequency responses of pitch rate and heading at point "a" in Fig. 1 to their respective primary controls at point "p" in Fig. 1 are shown in Figs. 6 and 7, respectively, for a mathematical model of the UH-60A rotorcraft, with a stability and control augmentation system (SCAS), at hover[12]. Barring additional delays, there is a potential phase margin of less than 40 deg at a crossover frequency of 3.0 rad/sec in Fig. 6 for pitch attitude control (subtract an additional 90 deg from the phase angles in Fig. 6), and a potential phase margin of less than 40 deg at a crossover frequency of 3.5 rad/sec in Fig. 7 for heading control. We next examine the effect of the frequency response of the visual system in the context of these potential crossover frequencies which govern the pilot's control bandwidth.

FREQUENCY RESPONSE OF THE VISUAL SYSTEM

The results of the CGI tests are presented in Figs. 8 through 10 for the pitch and yaw axes, which provide superior visual signals for the photometric sensor. The pitch axis was evaluated

with (Fig. 8) and without (Fig. 9) the CGI compensation scheme[8], using a recording sample period of 20 Hz. Without compensation, the throughput delay of the CGI is 57 ms, as shown by the phase lag in Fig. 9. Note that this delay is from point "a" to point "d" in Fig. 1. The delay from point "p" to point "a" is 55 ms, giving a total throughput delay from the pilot to the visual simulator of 112 ms. This additional 57 ms visual throughput delay reduces the potential phase margin at 3 rad/sec from less than 40 deg to less than 30 deg in Fig. 6 for pitch attitude control, and the potential phase margin at 3.5 rad/sec from less than 40 deg to less than 29 deg in Fig. 7.

There is no evidence of gain distortion in Fig. 9, but the data only go up to about 11 rad/sec. This is because the pilot had trouble generating inputs at frequencies above 2 Hz. The data shown in Fig. 8 demonstrate that the CGI compensation scheme does indeed correct the phase angle to zero, but there is a slight hint of gain distortion at about 13 rad/sec.

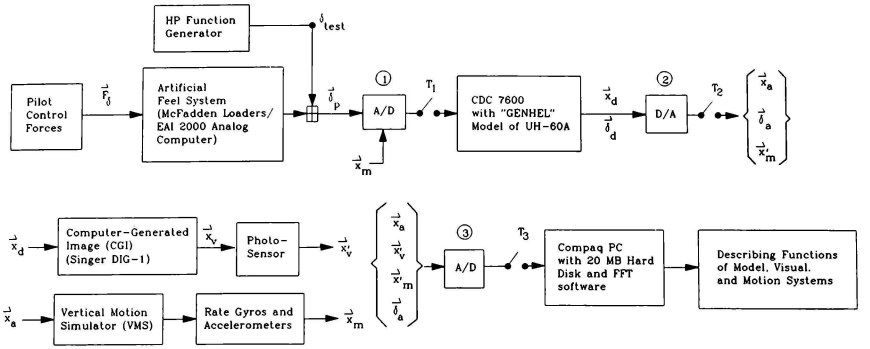
For the yaw axis evaluation (Fig. 10), the data recording sample rate was increased to 50 Hz, and the pilot was asked to concentrate on generating higher frequency inputs. With the CGI delay compensation scheme turned on, there is no evidence of phase lag accompanying a CGI delay and virtually no magnitude distortion up to frequencies of about 10 rad/sec. Above 10 rad/sec, there is a definite trend toward phase lead and gain amplification. This appears to be the "price" one has to pay for correcting the phase lag at lower frequencies.

CONCLUSIONS AND RECOMMENDATIONS

A novel technique has been used to measure the frequency response of the CGI visual generator used by the VMS at the NASA ARC. The results show that the CGI compensation scheme[8] can eliminate the phase lag due to pure time delay up to about 2 Hz, but above this frequency, the CGI response has phase lead and gain amplification. Only a limited amount of data was obtained above 2 Hz, because the inputs to the CGI were generated by a UH-60A test pilot, and this is probably the upper limit of excitation that can be generated by a human pilot. However, this frequency is believed to provide sufficient visual state bandwidth for flying qualities research on the VMS. It now remains to address the issues of visual content, including resolution and luminance contrast of image details, field of view, and visual perspective required for simulating a specific task with in-flight fidelity.

REFERENCES

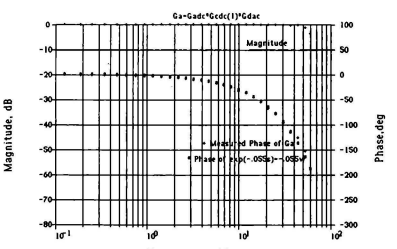
1. Heffley, Robert K., Warren F. Clement, et al., Determination of Motion and Visual System Requirements for Flight Training Simulators, ARI TR-546, August 1981.
2. Heffley, R. K., T. M. Schulman, and W. F. Clement, An Analysis of Airline Landing Flare Data Based on Flight and Training Simulator Measurements, NASA CR 166604, August 1982.
3. Ferguson, Samuel W., Warren F. Clement, William B. Cleveland, and David L. Key, "Assessment of Simulation Fidelity Using Measurements of Piloting Technique in Flight," Systems Technology, Inc., Paper No. 347, Presented at the 40th Annual Forum of the American Helicopter Society, Arlington, VA, May 1984.
4. Clement, Warren F., William B. Cleveland, and David L. Key, "Assessment of Simulation Fidelity Using Measurements of Piloting Technique in Flight," Systems Technology, Inc., Paper No. 348, Presented at the AGARD Guidance and Control Panel 38th Symposium, Helicopter Guidance and Control Systems for Battlefield Support, Monterey, CA, May 1984.
5. Ferguson, Samuel W., Warren F. Clement, Roger H. Hoh, and William B. Cleveland, "Assessment of Simulation Fidelity Using Measurements of Piloting Technique in Flight--Part II," Systems Technology, Inc., Paper No. 371, Presented at the 41st Annual Forum of the American Helicopter Society, Ft. Worth, TX, May 1985.
6. Ferguson, Samuel W., and Warren F. Clement, Assessment of Simulation Fidelity Using Measurements of Piloting Technique in Flight, Systems Technology, Inc., Technical Report No. 1184-1, August 1984, Revised March 1986.
7. Hoh, R. H., Investigation of Outside Visual Cues Required for Low Speed and Hover, Systems Technology, Inc., TR 1213-1, June 1984.
8. McFarland, R. E., CGI Delay Compensation, NASA TM 86703, January 1986.
9. Sinacori, J. B., Piloted Aircraft Simulation Concepts and Overview, Systems Technology, Inc., TR 1074-1, March 1978.
10. Jex, Henry R., and David G. Mitchell, The Stability & Control Identification of the Gossomer Human Powered Aircraft by Analysis and Flight Test, NASA CR 3627, October 1982.
11. Jex, H. R., R. E. Magdaleno, P. Gelhausen, and M. P. Tischler, "Pre-and Post-Flight-Test Models Versus Measured Skyship-500 Control Responses," AIAA Paper No. 87-2508, August 1987.
12. Howlett, J. J., "UH-60A Black Hawk Engineering Simulation Program," Sikorsky Aircraft Document No. SER 70452, December 1981.



NOTES:

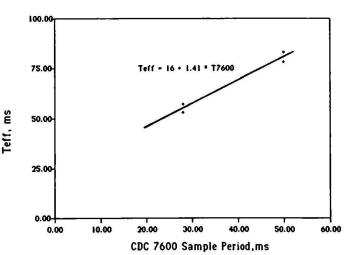
1. The A/D involves 16 bit, 10 volt analog-to-digital converters ("RIOU") and a PDP-11/55 that transfers data to the CDC 7600 via a digital-to-digital transfer ("CIOU"). The ADC sample period, T_1, is 28 ms.
2. The D/A involves a digital-to-digital transfer to a PDP-11/55 and a 16 bit, 10 volt digital-to-analog converter. The DAC sample period, T_2, is 28 ms.
3. The A/D is a 12 bit, 10 volt analog-to-digital converter. The ADC sample period, T_3, is 50 ms for most tests, but is set to 20 ms for the CGI tests.

Figure 1. Functional Block Diagram of Simulation Components and Data Measurement System



Notes:
1. Data are only for coherence values of 0.70 or greater.
2. Nyquist frequency is 157.0 rad/sec.
3. CDC 7600 sample period is 28 ms.

Figure 2. Frequency Response of G_a



Notes:
1. T_{eff} is from points "p" to "a" in Fig. 1.
2. Data is for 2 runs at each sample period.
3. Equation for T_{eff} is a best fit of the data.

Figure 3. Effective Throughput Delay of the VMS as a Function of the CDC 7600 Sample Period.

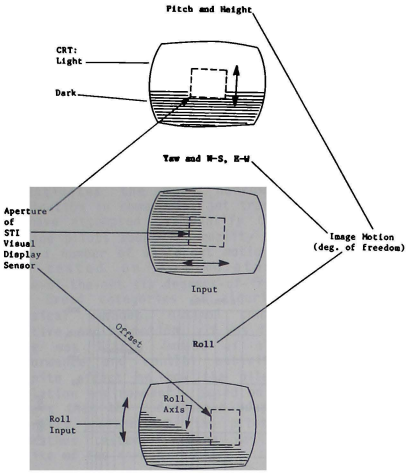


Figure 4. Views of a High Contrast Horizon Image on the Television Monitors Used to Sense Image Motions

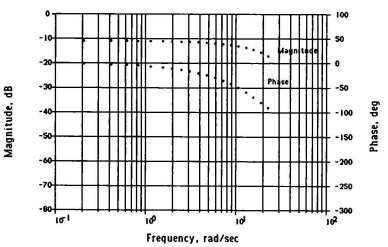
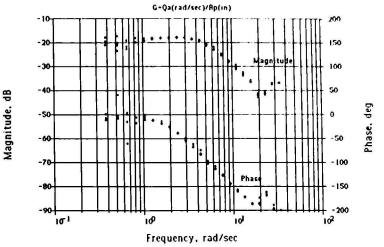
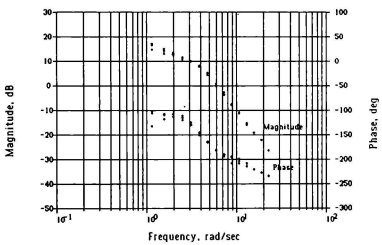


Figure 5. Frequency Response of Photometric Sensor, $G_{ps}(s=j\omega)$.



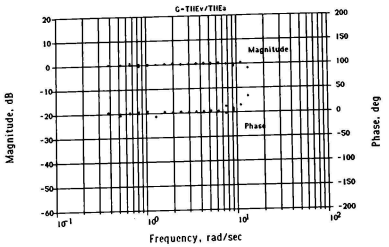
Notes: 1. Data are only for coherence values of 0.70 or greater.
2. Nyquist frequency is 62.8 rad/sec.
3. UH-60A is at hover.
4. Units are rad/sec/inch
5. Data are for four runs.

Figure 6. Frequency Response of q_a/B_p



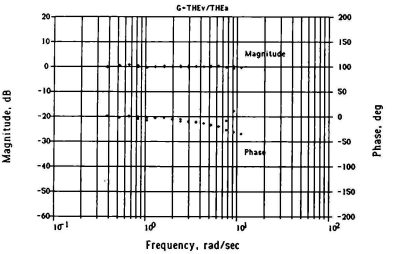
Notes: 1. Data are only for coherence values of 0.70 or greater.
2. Nyquist frequency is 157.0 rad/sec.
3. UH-60A is at hover.
4. Units are deg/inch.
5. Data are for thirty-two runs.

Figure 7. Frequency Response of Ψ_a / P_p



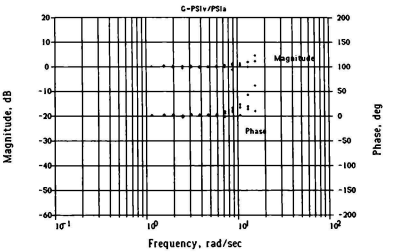
Notes: 1. Data are only for coherence values of 0.70 or greater.
2. Nyquist frequency is 62.8 rad/sec.
3. CGI compensation algorithm is on.
4. Photometric sensor dynamics have been removed from data shown.
5. Data are for two runs.

Figure 8. Frequency Response of θ_v/θ_a



Notes: 1. Data are only for coherence values of 0.70 or greater.
2. Nyquist frequency is 62.8 rad/sec.
3. CGI compensation algorithm is off.
4. Photometric sensor dynamics have been removed from data shown.
5. Data are for two runs.

Figure 9. Frequency Response of θ_v/θ_a



Notes: 1. Data are only for coherence values of 0.70 or greater.
2. Nyquist frequency is 157.0 rad/sec.
3. CGI compensation algorithm is on.
4. Photometric sensor dynamics have been removed from data shown.
5. Data are for three runs.

Figure 10. Frequency Response of ψ_v/ψ_a

THE RESPONSE OF AIRLINE PILOTS TO FLIGHT SIMULATOR MOTION

87-2436

Lloyd D. Reid* and Meyer A. Nahon†

The University of Toronto Institute for Aerospace Studies
Toronto, Ontario, Canada

Abstract

The use of physical motion in flight simulation is still a much debated topic. This paper investigates the more narrow issue of its application in commercial jet transport simulators. We have attempted to quantify the perceptions of airline pilots about the quality of motion possible when a number of different motion-drive algorithms are tested on a simulator employing a state-of-the-art six degrees-of-freedom motion-base. Four broad categories of algorithm were tested; classical washout, optimal control, coordinated adaptive, and no-motion. It was found that although there was little impact of algorithm type on performance and control activity, there was a definite effect on how the pilots perceived the simulation environment. Based on these findings it appears that the coordinated adaptive algorithm is generally preferred by the pilots over the other algorithms tested. There was almost unanimous dislike of the no-motion case.

Introduction

The present study was prompted by the need to install a motion-drive algorithm on the recently commissioned UTIAS Flight Research Simulator shown in Figure 1. It was felt that perhaps one of the more recent developments such as adaptive or optimal control algorithms would be best suited to such a research facility.

A review of recent reports covering the design and evaluation of motion-drive algorithms was carried out, keeping in mind that the present system is a six degrees-of-freedom synergistic motion-base with hydrostatic bearings. As might be expected, a perfect match between our needs and the reported material was not found although some very helpful information was located. For example, the work implemented in Reference 1 was carried out on a simulator employing hydrostatic bearings but was restricted to three degrees-of-freedom. Reference 2 utilized six degrees-of-freedom but involved the simulation of combat aircraft as did Reference 3 (but with five degrees-of-freedom). The studies reported in References 4 and 5 employed the simulation of a B737 and came closest to satisfying our requirements (although only five degrees-of-freedom were active). In fact the nonlinear coordinated adaptive washout routines from these latter two references formed the basis for the AW algorithms studied in the present project. The work reported in Reference 6 was found to be extremely helpful. It indicated that the adaptive washout was preferred by pilots over the classical fixed filter washout in a three degrees-of-freedom helicopter simulation. A comparison between an optimal control and a classical washout is presented in Reference 7 where no significant differences in pilot preference were found in a two degrees-of-freedom simulation of a VTOL aircraft.

Following this review it was decided to carry out a study to obtain pilot evaluations of the motion quality produced by classical washout, optimal control and adaptive washout using the full six degrees-of-freedom of our motion-base. It was anticipated that the findings would also help to clarify the relative merits of the various alternatives for commercial jet transport training simulators.

Description of the UTIAS Flight Research Simulator

The motion-base of the UTIAS Flight Research Simulator is a CAE Series 300 six degrees-of-freedom synergistic unit incorporating hydrostatic bearings. Its performance characteristics are fully documented in Reference 8. Its signal-to-noise properties and dynamic response are equal to or better than most current commercial systems. A DC-8 cab is mounted on the motion-base and the whole system is run at a 20 Hz update rate by a Perkin Elmer 3250 digital computer. Engine and aerodynamic sounds are generated by a digital system controlled by the computer. The other major components are outlined below. A detailed description is contained in Reference 9.

Aircraft Equations

The simulated aircraft was a Boeing 747. The flight equations were based on References 10 to 12. The aerodynamic forces and moments were obtained from Reference 11 and stored in the form of lookup tables. The equations were solved using a 2nd order Adams-Bashforth numerical integration scheme. A full set of ground handling equations was developed as well, along with a JT9D-3 engine model derived from Reference 11.

Navigation and Landing Aids

Navigation and landing aids were generated to represent an airport terminal area. The runway had an ILS (instrument landing system) with a 3° glideslope. This was sufficient to allow the pilots to complete the flying tasks associated with this study.

Visual Display System

The forward out-the-window CRT display is viewed through a collimating optical unit employing a beam splitter and a mirror (from a VITAL II system). The field of view is 40° horizontally and 30° vertically. The monochromatic image (yellow) is produced by a vector generator driven by the Perkin Elmer 3250 digital computer and consists of straight line segments on a dark background. This system is

*Professor and Associate Director, Associate Fellow AIAA
†Project Engineer

used to produce a head up display representing the outside world in perspective as shown in Figure 2. The display is updated at 20 Hz.

The ground plane is represented by a grid of squares and horizon glow is also included. A set of three T-bars along the approach to the runway are provided as a visual aid to landing. When the aircraft is on the localizer and glideslope all the T-bars are aligned and the pilot should attempt to fly through their cross-piece intersection points. A set of pole indicators beside the runway act as a visual aid during the flare portion of a landing maneuver. This display was quite natural to use and the pilots had no difficulty performing their flying tasks while looking out the window.

Turbulence

Turbulence effects are included in the simulation through the wind velocity at the aircraft's center of mass and the wind gradient in the spanwise direction. The effect of turbulence on the horizontal tail is accounted for by incrementing its angle-of-attack by a time delayed version of the increment in the wing's angle-of-attack due to turbulence (i.e., the frozen flow assumption). In all, three components of turbulence and two of turbulence gradient are generated as independent non-Gaussian processes. The technique employed is the one described in References 13 to 15. The resulting patchy turbulence was fairly realistic. The intensity of the turbulence was reduced smoothly from moderate to zero near the runway so as not to disturb the aircraft during the final phase of the landing approach.

Buffet and Runway Roughness

The buffet vibration is intended to be representative of the buffet felt in an aircraft as a result of the flaps and landing gear being exposed to the slipstream. The runway roughness is intended to indicate to the pilot that the wheels of the aircraft are making ground contact. Only very simple sinusoidal waveforms are employed and no attempt is made to duplicate actual motion time histories. The signals are fed directly to the motion-base without passing through the washout filters and hence are unaffected by changes to the washout algorithms. The same motion increment is applied to all six actuators thereby producing primarily a heave acceleration.

Motion and Visual Cue Timing

The relative time between the various steps in the simulation process is important in determining the quality of the simulation as perceived by the pilot. In the present instance the complete sequence of events corresponding to one iteration cycle takes place in 50 ms. The overall time delay sensed by the pilot depends upon both the software and the hardware. The out-the-window visual display and the motion drive command signals are both generated by the Perkin Elmer 3250 computer. Based on measured execution times for the software and dynamic response measurements for the motion-base the following time delay estimates were obained for two scenarios. Under normal operating conditions with smooth continuous inputs, the additional time delays between a pilot input on the controls and a visual display or motion response, beyond that due to the aircraft equations of motion (allowing 25 ms of delay to represent an average value for the time-delay in sampling the pilot's input), are:

Visual time delay	-	14-24 ms
Motion time delay	-	0-50 ms

In the case where a discontinuous step input by the pilot is assumed to be the test signal, these time delays are increased:

Visual time delay -	114-124 ms
Motion time delay -	
classical and optimal algorithms	60-110 ms
adaptive algorithm	160-210 ms

It is felt that no significant time delay effects were experienced in the present study.

Motion-Base Drive Algorithms

Three types of motion-base drive algorithms were studied in the present project (in addition to a no-motion case):

(1) classical washout of the type currently employed in airline flight simulators (References 16 to 18),
(2) optimal control (Reference 19),
(3) coordinated adaptive (Reference 5).

Our interest was in adapting them for use on the UTIAS Flight Research Simulator and in obtaining an unbiased assessment of the quality of the motion they produce. The details of the exact form of the resulting algorithms are presented in References 20 and 21. A brief description outlining their key features is given below.

Classical Washout

In the classical washout algorithm, fixed coefficient high-pass filters are used to prevent low frequency linear acceleration signals from reaching the motion-base. This is done because it is these low frequency signals that can cause the simulator motion system to exceed its physical displacement limits. The same process is used in the yaw degree-of-freedom mainly to wash out the large yaw angles associated with steady turns. A process known as tilt-coordination is used to generate simulator low frequency pitch and roll angles in response to low frequency aircraft longitudinal and lateral specific force. This aligns the simulator pilot relative to the local gravity vector so that his vestibular system senses a resultant specific force with the same relative orientation as that sensed by the pilot in the aircraft. If done properly, this can be used to create the illusion of sustained longitudinal and lateral acceleration. To be successful, the simulator's angular tilt rates must go undetected by the pilot during this process. For this reason tilt rate limiting of 3°/s is employed in the present study. Another feature of the crossfeed from aircraft specific force to simulator tilt is that during a coordinated turn, the simulator bank angle generated as the initial aircraft roll-rate begins, is washout out, returning the simulator to a level

condition thereby generating the correct sensation of zero sustained lateral specific force during the turn. A combination of the low frequency crossfeed and high-pass filters applied to the aircraft pitch and roll rates tends to produce an unfiltered overall simulator response to uncoordinated aircraft pitch and roll. In the case of a jet transport simulation, because the actual aircraft pitch and roll displacements are relatively small this causes no serious problem.

Optimal Controller

Like classical washout, the optimal controller algorithm uses fixed coefficient high-pass filters and low-pass crossfeed filters to restrict the motion of the flight simulator. Two significant differences are the use of optimal control theory to select the form of the filters and the optimal controller's attempt to match the pilot's vestibular sensations in the aircraft and in the simulator. The latter requires the use of a linear mathematical model of the vestibular system. The filters are similar in function to those in the classical washout. However, it was found that rate limiting could not be used on the tilt-coordination crossfeed because this caused excessive bank angles in the simulator during coordinated turns. This results from the use of lateral acceleration in the crossfeed channel by the optimal controller (classical washout uses lateral specific force).

Coordinated Adaptive Washout

This algorithm is somewhat similar in general structure to the classical washout. The significant difference is the use of an adaptive control scheme to continuously adjust the parameters in the filters in response to the current state of the simulator motion-base. As the motion limits are approached the filters become more restrictive. This allows the use of fairly modest filtering during the rest of the time thereby improving the overall quality of motion. It is found that care must be taken in selecting the system gains and in limiting the range of the adaptive parameters in order to avoid instabilities.

Selecting Algorithm Parameters

In order to generate a range of motion-base drive algorithms for testing, three parameter sets were generated for each algorithm type. Complete details are contained in References 9, 20 and 21. The tuning process involved choosing washout filter parameters which would yield a range of simulator motions, from the most active (while still remaining within the limits of the motion-base of the UTIAS Flight Research Simulator) to the least active, for a given set of aircraft maneuvers.

The classical washout filters were tuned by modifying the filter characteristics. In all cases a scale factor of 0.5 was applied to the motion variables coming from the flight equations. The most active version, CW1, was chosen to produce simulator motions close to the maximum actuator travel available while responding to the three design maneuvers. The second set, CW2, had the same order filters as CW1 but was tuned to have a more restricted low frequency response. The third set, CW3, was chosen to be even more restrictive and had all its high-pass filters increased by one order.

In the case of the optimal controller the same scale factor of 0.5 was used at the input. The most active version, OC1, was created by adjusting the weights of its cost functional to achieve the same level of response as CW1 to the design maneuvers. The second set, OC2, was taken to be the same as OC1 except that it was altered to allow more simulator roll. The third set, OC3, was obtained by starting with OC1 and increasing the penalty in the cost functional associated with simulator motion, thereby creating a more restrictive filter.

The coordinated adaptive washout algorithms were tuned by starting with input scale factors of 1.0, 0.5 and 0.25 for AW1, AW2 and AW3 respectively. The fixed algorithm parameters for AW1 were then selected to give peak simulator motion similar to CW1 for the test maneuvers. The parameters for AW2 were selected to give peak responses similar to CW2. In addition, an attempt was made in going from AW1 to AW2 to reduce the false lateral specific force cue in roll maneuvers due to excessive simulator roll and its slow return to the neutral position. AW3, the most restrictive filter set of the three, was identical to AW2 except for the above mentioned scale factors.

Experimental Design

Because the purpose of this study was to assess the suitability of motion-base drive algorithms for use in jet transport flight simulators, it was decided to employ current airline pilots in the evaluation process. The primary assessment consisted of having the pilots fly a flight sequence in the UTIAS Flight Research Simulator and then rate the quality of motion. This was repeated for 10 motion-base drive algorithms consisting of the 9 mentioned above and a no-motion case (designated as NM) in which only buffeting and runway roughness were present.

The flying sequence consisted of the following items (see Figure 3):

(1) Heading and altitude hold in turbulence.
(2) VOR intercept.
(3) Deceleration while tracking a VOR radial.
(4) Descent.
(5) Sidestep maneuver to capture an ILS.
(6) ILS approach to touchdown.
(7) Takeoff and climb-out including an engine failure.
(8) Wheel and rudder induced transients.

The 7 airline pilots taking part in this study were unpaid volunteers. Their participation was an expression of their professional interest in improving the effectiveness of flight simulators. Table 1 summarizes their flying and simulator experience. It was emphasized to the pilots that they were only to judge the quality of the motion cues and not any other aspects of the simulation. It was also made clear to them that they should rate the simulator motion relative to that which would be experienced in an actual aircraft and not relative to that experienced in their airline flight simulators.

Both subjective and objective measurements were used to determine the impact of the motion-base drive algorithms on the pilots. The subjective measurements consisted of the two rating scales contained in Figures 4 and 5. In addition the pilots were encouraged to add any comments they wished. The rating scale of Figure 4 was developed at UTIAS. It is based on work reported in Reference 22 and the adjectives appearing on the scale are spaced so as to produce an equal interval scale. Immediately following each trial the pilots were asked to mark on each vertical line their assessment of the quality of motion associated with their control inputs on the column, wheel, rudder pedals and throttle and that produced by the turbulence inputs and ground contact. The rating scale of Figure 5 was developed at MIT and reported in Reference 7. Here the pilot must give a numerical rating.

The objective measurements covered the pilot's control activity, the performance of the flying tasks and the motion of the flight simulator.

Training took place with the simulator motion-base completely shut down (i.e., no buffet or runway roughness present). The pilots were allowed to practise the flying sequence until they felt proficient in the assigned tasks. Typically 2 hours of flying were logged during training in a single morning or afternoon session. Next the evaluation trials were carried out. The 10 motion-base drive algorithms were assigned to each of the 7 pilots in a randomized order. Five flights were performed in a single session by each pilot and thus two sessions per pilot were required. Each session lasted approximately 2.5 hours. Only one morning or afternoon session per day per pilot was allowed.

Results, Analysis and Discussion

Simulator Motion

As expected, the simulator motion can be strongly influenced by the motion-base drive algorithm. In the present case an analysis of variance indicated that there was a significant (0.1%) effect on average x and z specific force and RMS actuator length (see Figure 6), and on all the standard deviations of specific force $\underline{f}$ and angular velocity $\underline{\omega}$ (data analyzed with NM deleted).

Pilot's Control Activity and Performance

In general it was found that there was no influence of motion drive algorithm type on the pilot's control activity and performance. The greatest variation in performance was noted in the touchdown rate of descent (see Figure 7 for example).

Pilot Ratings

In spite of the lack of influence of the motion-base algorithms on such measures as control activity and task performance, there was a definite strong impact on pilot opinion. This was reflected in pilot comments (fully documented in Reference 9), and pilot ratings. Figure 8 shows the summary of pilot ratings of simulator response to wheel inputs using the UTIAS scale. Figure 9 does the same for the overall rating item included on the MIT scale. It can be seen that the mean values of the ratings (averaged over 7 pilots) are affected by the motion-base drive algorithm form. An analysis of variance was performed on the results from each rating scale. Table 2 shows the results for the UTIAS scale (the MIT scale produced similar values). It is seen that treatment (algorithm) effects are highly significant. In order to highlight the individual items contained in both rating scales an analysis of variance was performed on subsets of the data corresponding to each item in isolation. The results are summarized in Tables 3 and 4 as the probability corresponding to the computed F ratio. It is interesting to note that a repeat of this analysis with the NM data deleted produced essentially the same results. In order of decreasing significance it was found for the UTIAS scale that the motion-base drive algorithm type affects pilot ratings of Ground Contact, Turbulence, Column, Wheel, and Throttle with little impact on Pedals. For the MIT scale the corresponding sequence was Smoothness, Amplitude, Sense, Overall, and Phase Lag with little impact on Disorientation and Discomfort.

Table 5 summarizes the rank order of the ratings based on the average results. Cases with identical mean ratings are joined together by underlining them. It can be seen that the sequences are somewhat variable across the rated attributes. Despite this it is possible to see some trends. For example:

(1) NM appears most often towards the extreme right hand side of the array, indicating that the no-motion algorithm is not very well liked. (The Smoothness case where NM is ranked first is not relevant.)

(2) AW2 appears most often towards the extreme left hand side of the array, indicating that it produced the best motion quality in most cases. This is borne out by its ranking as number one under the attribute "Overall". (The Amplitude case where AW2 is ranked towards the right is only an indication of the amount of motion perceived. From Figure 5 it is seen that its average value of 2.6 was near 3.0 which indicates a reasonable amount of motion.)

(3) In the "Overall" attribute rating CW2 is ranked second to AW2. This is consistent with the performance of CW2 in the other categories.

(4) The best of the optimal controller algorithms is OC2. Its performance is found to be, on average, slightly to the right of centre in Table 5. However OC2 was ranked lower than CW2 and AW2 in almost every category.

(5) From the "Overall" attribute rating it is seen that all the optimal controller algorithms are ranked below all the classical and adaptive algorithms. The classical and adaptive rankings are intermixed.

Pilot Consistency

The consistency of the pilots in assigning subjective ratings is always a concern. Some idea

of this for the present group of pilots can be obtained from the standard deviations of the data plotted in Figures 8 and 9. A series of tests reported in Reference 9 and involving the same group of pilots immediately following the tests reported herein (pilots 1 to 6 in Table 1) can be used to shed further light on their consistency. In these paired comparison tests three short flight segments were used each taking 4 minutes of flying time. Segment 1 involved deceleration and descent (items 3 and 4 of Figure 3), Segment 2 involved an ILS approach and takeoff (items 6 and 7 of Figure 3 but without the engine failure) and Segment 3 involved test maneuvers (item 8 of Figure 3 plus an engine failure). Each of the three segments was studied in a block of experimental trials. A single trial consisted of flying the particular segment twice in close succession but with different motion-base drive algorithms in place. After flying each pair the pilot was asked to indicate which algorithm generated the better motion quality. The motion algorithm cases tested were CW2, OC2, AW2 and NM. All possible pairs were presented to each pilot using a randomized Latin-Square design.

Based on the results of these tests, the rank orders produced by each pilot are presented in Table 6. In three cases a pilot was inconsistent in ordering one pair of trials out of the six pairs employed and this results in three equally probable rank orderings for that Flight Segment. In 15 out of 18 cases the pilots were completely internally consistent with each of their 6 paired rankings agreeing completely with the displayed sequence. However, they display a wide range of preferences depending upon the flight segment. The between pilot variation can be quantified by using the coefficient of concordance W defined in Reference 23 which goes from 1 to 0 as the inter-pilot consistency goes from perfect agreement to a total lack of agreement. In the present case values of W = 0.144, 0.211 and 0.189 for Segments 1 to 3 respectively indicate very poor agreement among pilots.

Based on our sample of the airline pilot population it appears that in rating motion:

(1) pilots are very self-consistent,
(2) pilots may prefer different motion algorithm properties when carrying out different maneuvers,
(3) inter-pilot differences are quite large.

Conclusions and Recommendations

(1) In the present study the motion-base drive algorithms had almost no impact on flying performance and control activity.

(2) The pilots preferred physical motion to be present in the simulator. They felt that it added to the realism of the simulation and was helpful in the piloting task.

(3) In general, each pilot was fairly consistent in his ratings of the various motion-base drive algorithms.

(4) There was considerable variability among pilots in the rating process. This is demonstrated by the pilot comments and the small values found for the coefficient of concordance (W) in the paired comparison test analysis.

(5) In spite of the variability reported in (4) above, the trends in pilot ratings caused by the different motion-base drive algorithms were significant in the cases of the attributes Column, Wheel, Throttle, Turbulence, Ground Contact, Smoothness, Sense, Amplitude, Phase Lag and Overall. There were no significant trends noted for Rudder Pedals, Discomfort and Disorientation.

(6) Under the attribute Overall, the pilots ranked the motion-base drive algorithms as follows (from best to worst): AW2, CW2, CW3, AW1, CW1, AW3, OC2, OC1, OC3, NM. However the other pilot rating responses indicate that the algorithm sequence for any one of the rated attributes may differ from this in detail.

(7) As indicated by the pilot comments and the pilot ratings, it is possible for the simulator motion amplitude to be judged as too great even though it is well below that expected in an actual aircraft.

(8) No instances of simulator sickness were observed although there were several complaints of disorientation in the no-motion (NM) runs.

(9) Based on the good performance of the coordinated adaptive washout algorithm it appears worthwhile to investigate its further improvement through a systematic study of alternate forms for its cost function. As well, more sophisticated cost functions should be used to give the algorithm more 'intelligence' to handle specific situations.

(10) Motion-base drive algorithms should be selected to suit

(i) the individual pilot,
(ii) the individual degree-of-freedom,
(iii) the individual maneuver,
(iv) the particular simulator.

A means of dynamically achieving this goal should be developed and tested.

Acknowledgements

The work reported in this paper was performed during the final year of a three year contract held with the Canadian Transportation Development Centre (Contract No. OSD83-00101, Project No. 6069). The authors would like to thank all the evaluation pilots who took part in the project. Without their dedicated efforts this study could not have been completed. In addition we would like to thank the staff of Air Canada and the Canadian Airline Pilots Association who helped us with the fine tuning of the simulator and the recruitment of the evaluation pilots.

The development of the simulator employed in this study was funded by the Canadian Natural Sciences and Engineering Research Council and the Ontario Government.

References

1. Hosman, R.J.A.W., van der Vaart, J. C. and van de Moesdijk, G. A. J., "Optimization and Evaluation of Linear Motion Filters," Delft University of Technology, Dept. of Aerospace Engineering, Memorandum M-326, March 1979.

2. Irish P. A. and Brown, J. E., "Subjective Motion Discrimination in the Simulator for Air-to-Air Combat," AFHRL-TR-78-26, August 1978.

3. Hofmann, L. G. and Riedel, S. A., "Manned Engineering Flight Simulation Validation, Part I: Simulation Requirements and Simulator Motion System Performance," AFFDL-TR-78-192, Part I, February 1979.

4. Parrish, R. V. and Martin, D. J. Jr. "Application of Nonlinear Adaptive Motion Washout to Transport Ground-Handling Simulation," NASA TM 84568, February 1983.

5. Parrish, R. V. and Martin, D. J. Jr., "Comparison of a Linear and a Nonlinear Washout for Motion Simulators Utilizing Objective and Subjective Data from CTOL Transport Landing Approaches". NASA TN D-8157, June 1976.

6. Bitner, M. E., "Investigation of Motion Base Drive Techniques," NADC-77306-20, March 1978.

7. Bussolari, S. R., Sullivan, R. B. and Young, L. R., "Vestibular Models for Design and Evaluation of Flight Simulator Motion," presented at the Royal Aeronautical Society Conference on Advances in Flight Simulation Visual and Motion Systems, London, April 1986.

8. Grant, P. R., "Motion Characteristics of the UTIAS Flight Research Simulator Motion-Base," University of Toronto UTIAS TN No. 261, July 1986.

9. Reid, L. D. and Nahon, M. A., "Flight Simulation Motion-Base Drive Algorithms: Part 3 - Pilot Evaluations," University of Toronto, UTIAS Report No. 319, December 1986.

10. Leung, Y.M., "Solution of the General Flight Equations in Real Time," M.A.Sc. Thesis, University of Toronto, UTIAS, 1985.

11. Hanke, C.R., and Nordwall, D.R., "The Simulation of a Jumbo Jet Transport Aircraft, Volume II: Modeling Data," NASA CR-114494, 1970.

12. Hanke, C.R., "The Simulation of a Large Jet Transport Aircraft Volume I: Mathematical Model," NASA CR-1756, March 1971.

13. Gerlach, O.H. and Baarspul, M., "Calculation of the Response of an Aircraft to Random Atmospheric Turbulence. Part II: Asymmetric Motions," Technological University of Delft, Dept. of Aeronautical Engineering, Report VTH-139, April 1968.

14. van de Moesdijk, G.A.J., "The Description of Patchy Atmospheric Turbulence, Based on a Non-Gaussian Simulation Technique," Delft University of Technology, Dept. of Aeronautical Engineering, Report VTH-192, February 1975.

15. van de Moesdijk, G.A.J., "Non-Gaussian Structure of the Simulated Turbulent Environment in Piloted Flight Simulation," Delft University of Technology, Dept. of Aerospace Engineering, Memorandum M-304, April 1978.

16. Baarspul, M., "The Generation of Motion Cues on a Six-Degrees-of- Freedom Motion System,"Delft University of Technology, Dept. of Aerospace Engineering, Report LR-248, June 1977.

17. Parrish, R. V., Dieudonne, J. E. and Martin, D. J. Jr., "Motion Software for a Synergistic Six-Degree-of-Freedom Motion Base," NASA TN D-7350, December 1973.

18. Schmidt, S. F. and Conrad, B., "Motion Drive Signals for Piloted Flight Simulators," NASA CR-1601, May 1970.

19. Sivan, R., Ish-Shalom, J. and Huang, J.-K., "An Optimal Control Approach to the Design of Moving Flight Simulators," IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-12, No.6, Nov./Dec. 1982, pp. 818-827.

20. Reid, L. D. and Nahon, M. A., "Flight Simulation Motion-Base Drive Algorithms: Part 1 - Developing and Testing the Equations," UTIAS Report No. 296, December 1985.

21. Reid, L. D. and Nahon, M. A., "Flight Simulation Motion-Base Drive Algorithms: Part 2 - Selecting the System Parameters," University of Toronto, UTIAS Report No. 307, May 1986.

22. McDonnell, J. D., "An Application of Measurement Methods to Improve the Quantitative Nature of Pilot Rating Scales," IEEE Transactions on Man-Machine Systems, Vol. MMS-10, No.3, Sept 1969, pp.81-92.

23. Seaver, D. A. and Stillwell, W. G., "Procedures for Using Expert Judgement to Estimate Human Error Probabilities in Nuclear Power Plant Operations," U.S. Nuclear Regulatory Commission, NUREG/CR-2743, 1983, pp. A-1 to A-12.

Table 1. Pilot Experience

SUBJECT NUMBER	CURRENT POSITION	TOTAL FLYING HOURS	TRANSPORT FLYING HOURS	FLIGHT SIMULATOR HOURS
1	CAPTAIN DC9	11,000	5,500	500
2	CAPTAIN DC9	14,300	11,400	350
3	[a]F/O DC9	10,000	5,000	200
4	F/O L1011	5,000	4,000	250
5	F/O DC9	5,000	150	250
6	F/O DC9	5,000	4,000	150
7	[b]S/O B727	5,500	350	180

[a]F/O - First Officer
[b]S/O - Second Officer

Table 2. Analysis of Variance for the Complete Set of Results Using the UTIAS Rating Scale

Effect[a]	Degrees of Freedom	Sum of Squares	F Value	P(x>F)
Subjects	6.	44.9026	6.9634	<0.0001
Treatments	9.	220.1021	2.7553	<0.0001
Subjects× Treatments	54.	213.6717	3.6817	<0.0001
Variables	5.	34.8173	6.4793	<0.0001
Subjects× Variables	30.	45.5486	1.4127	0.0810
Treatments× Variables	45.	81.3420	1.6819	0.0067
Residual	270.	290.1770		
Total	419.	930.5613		

[a]"Treatments" refers to the 10 motion-base drive algorithms under study.
"Variables" refers to the 6 separate areas that are rated by the pilots.

Table 3. Analysis of Variance Summary for the UTIAS Rating Scale P(x>F)%

Subject Effects		
	Column	87.4
	Wheel	97.7
	Pedals	13.6
	Throttle	0.8
	Turbulence	1.3
	Ground Contact	1.0
Treatment[a] Effects		
	Column	0.2
	Wheel	5.2
	Pedals	29.7
	Throttle	6.6
	Turbulence	<0.1
	Ground Contact	<0.1

[a]Motion-Base Drive Algorithm

Table 4. Analysis of Variance Summary for the MIT Rating Scale P(x>F)%

Subject Effects		
	Smoothness	6.4
	Sense	<0.1
	Amplitude	<0.1
	Phase Lag	<0.1
	Discomfort	<0.1
	Disorientation	<0.1
	Overall	46.5
Treatment[a] Effects		
	Smoothness	<0.1
	Sense	0.6
	Amplitude	<0.1
	Phase Lag	4.1
	Discomfort	26.9
	Disorientation	16.1
	Overall	0.7

[a]Motion-Base Drive Algorithm

Table 5. Summary of Average Pilot Ratings (Best → Worst)

Column	AW2	CW1	CW2	AW1	OC2	CW3	AW3	OC3	OC1	NM
Wheel	AW2	CW2	OC2	AW1	CW3	CW1	AW3	OC3	OC1	NM
Pedals	CW2	AW2	CW1	AW1	CW3[b]	OC3	OC2	OC1	AW3	NM
Throttle	AW2	CW1	CW2	AW1	OC2	OC3	OC1	AW3	CW3	NM
Turbulence	AW2	CW2	AW1	CW3	CW1	AW3	OC2	OC1	NM	OC3
Ground Contact	AW1	CW1	AW2	CW3	OC1	CW2	OC3	OC2	AW3	NM
Smoothness	NM	AW3	CW3	AW2	OC2	CW2	AW1	CW1	OC1	OC3
Sense	AW2	CW3	CW2	OC2	AW3	NM	CW1	OC1	OC3	AW1
Amplitude[a]	OC3	AW1	CW1	OC1	CW2	OC2	AW2	CW3	AW3	NM
Phase Lag	AW2	CW3	CW2	OC2	NM	AW3	CW1	OC3	AW1	OC1
Discomfort	AW2	AW3	CW2	CW3	OC2	NM	OC1	AW1	CW1	OC3
Disorientation	CW1	CW2	AW2	AW3	CW3	OC2	OC3	AW1	NM	OC1
Overall	AW2	CW2	CW3	AW1	CW1	AW3	OC2	OC1	OC3	NM

[a]Most → Least
[b]Underlining joins identical averages

Table 6. Rating Sequences from Paired Comparison Tests

Pilot	Possible Sequences (BEST → WORST)		
	Segment 1	Segment 2	Segment 3
1	NM-AW2-CW2-OC2	CW2-AW2-NM-OC2	AW2-NM-OC2-CW2
2	NM-AW2-OC2-CW2	OC2-AW2-CW2-NM	OC2-NM-AW2-CW2
3	CW2-AW2-OC2-NM	CW2-AW2-NM-OC2	NM-AW2-OC2-CW2
4	CW2-AW2-OC2-NM	CW2-AW2-OC2-NM OC2-CW2-AW2-NM AW2-OC2-CW2-NM	AW2-CW2-OC2-NM
5	CW2-OC2-AW2-NM	CW2-OC2-AW2-NM OC2-AW2-CW2-NM AW2-CW2-OC2-NM	CW2-AW2-OC2-NM
6	CW2-AW2-OC2-NM OC2-CW2-AW2-NM AW2-OC2-CW2-NM	AW2-OC2-CW2-NM	OC2-CW2-AW2-NM

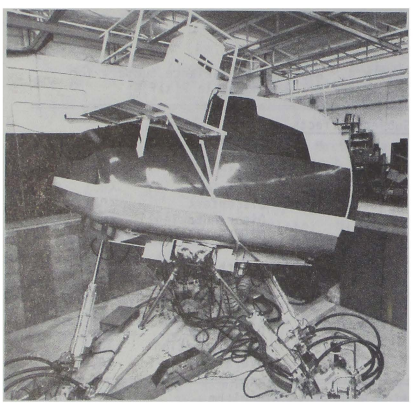


FIGURE 1 UTIAS FLIGHT RESEARCH SIMULATOR

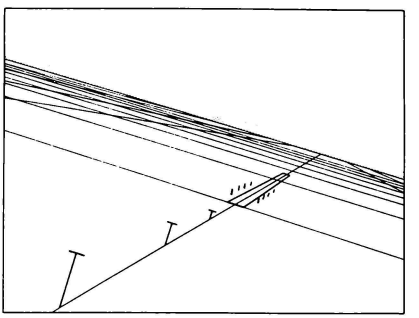


FIGURE 2 HEAD UP DISPLAY

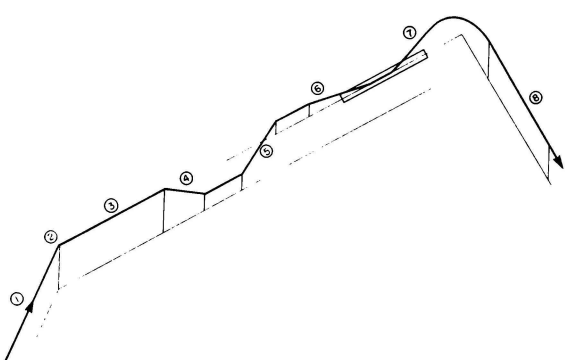
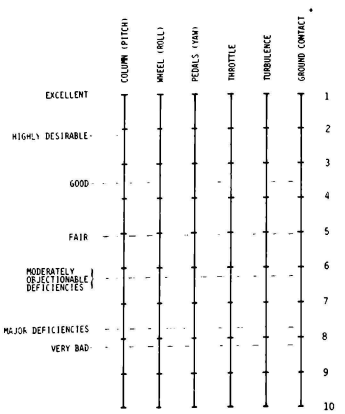


FIGURE 3 COURSE



* Includes touchdown bump, braking and steering.

FIGURE 4 UTIAS RATING SCALE

The motion of the simulator will be rated on the following seven scales:

Attribute	Rating 1	Rating 5	Response
SMOOTHNESS:	extremely smooth-comparable with fixed base	extremely jerky limit of tolerance	
SENSE:	definitely correct as in aircraft	totally reversed	
AMPLITUDE:	no motion experienced	at least twice that expected	
PHASE LAG:	none experienced	at least 180°	
DISCOMFORT:	none experienced	cannot continue maneuver	
DISORIENTATION:	none experienced	cannot perform maneuver	
OVERALL:	excellent	extremely poor	

FIGURE 5 MIT RATING SCALE

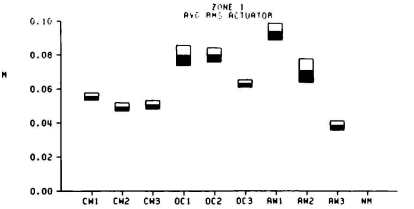


FIGURE 6 AVERAGE RMS ACTUATOR LENGTH

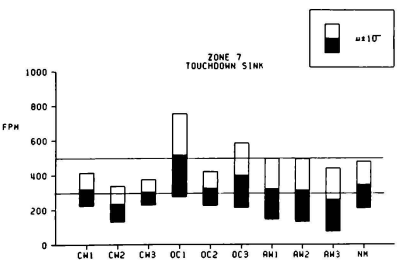


FIGURE 7 TOUCHDOWN RATE OF DESCENT

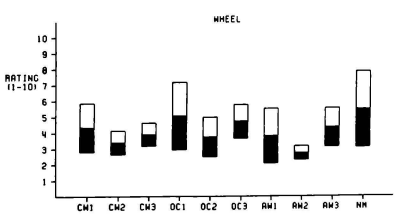


FIGURE 8 PILOT RATINGS OF RESPONSE TO WHEEL INPUTS

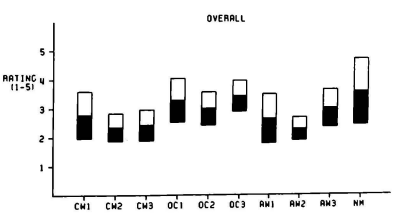


FIGURE 9 OVERALL PILOT RATINGS

87-2437

OPERATING EXPERIENCE OF A SMALL SIX AXIS MOTION SYSTEM INSIDE A DOME WITH A WIDE ANGLE VISUAL SYSTEM

A.G. Barnes
British Aerospace, Warton, UK.

Abstract

Operating Experience of a small six axis motion system inside a dome with a wide-angle visual display

A new research simulator at British Aerospace, Warton utilises a McFadden six degree of freedom motion system, operating inside a 30 foot diameter dome. Computer Generated Images giving a wide field of view are displayed on the dome by fixed projectors. This novel configuration has clear performance advantages over other solutions. As well as providing an uncluttered cockpit, it utilises proven technology. Details of the installation are described, and the configuration trade-offs are discussed.

1. Introduction

The widespread application of flight simulators to all spheres of aeronautical activity are an endorsement of the technology currently in use. Even so, there remain areas where flight simulation cannot be advocated as a substitute for flight. Similarly, the confidence and enthusiasm which characterises those people who are professionally involved in simulation does not extend throughout the aeronautical community. The gospel is spreading, and the message becomes clearer and more convincing as better technology is introduced. Nowhere is greater progress to be seen than in the field of visual displays for use in flight simulators.

Reproduction of the visual scene from the cockpit employs two elements: the image generator, and the display device. The use of physical models for image generation has been superceded almost completely by computer generated images (CGI). Remarkable progress has been made in the past decade, both in component technology, and in its application to system design. From a user's standpoint, the capability of CGI is dictated by cost - the customer will run out of money before the CGI vendor runs out of ideas for the generation of more complex images.

The situation concerning methods of display is less clear. Such variables as field of view, resolution, brightness, and up-date rate impose severe requirements. Depending on the application, various solutions are advocated. For example, air combat simulators use multi-projectors inside a dome, and civil transport simulators use TV monitors with beam splitter optics for collimation. More recently, in the latter application, systems like Rediffusion's 'Wide', which use infinity optics and projectors to give an uninterrupted horizontal view, are preferred. Although these simulators perform their respective training tasks adequately, in absolute terms, they fall far short of reality. Effort and expense are going into new display devices, to provide large field of view, high resolution, and high brightness, by exploiting the non-linear characteristics of the eye. One avenue of exploration is to project in a dome a high resolution insert onto a low resolution background, and slave the insert to the pilot's eye. Another category attempts to obviate the need for the dome, by mounting elements of the display system on the pilot's helmet.

Methods of providing motion cues to the pilot have been studied for many years. Devices have been incorporated into simulators to represent translations in three orthogonal axes, and rotations about those axes. Much has been published about the required performance, in terms of accelerations, rates, displacment and frequency response. Studies have been made to compare the sensors in the body - vestibular and proprioceptive - expressed in engineering terms - with the loop closures that are made by the pilot in a control task. Many simulator activities have been conducted successfully without motion cues. In many other cases, motion cues are mandatory.

Civil transport pilot training is one such case. Most simulators for this role utilise a large amplitude six degree of freedom synergistic motion system, driven by high quality hydraulic jacks with hydrostatic bearings to minimise friction. The motion platform is designed to carry the cockpit and the visual system. The algorithms used to drive the motion system have been long established. One of the requirements is that the pilot must not feel a sense of conflict between the impressions he receives from the visual cues and the motion cues. Ideally, the cues re-inforce one another, as in the aircraft. The designers of simulators currently in operation have tailored the motion system drive laws within certain performance limits, to suit the visual display.

The system to be described represents a new initiative in the integration of visual and motion cues. The visual system is not carried by the motion system, so that there is a two-way interaction: in some aspects, the motion system dictates the visual system drives, and in others, the visual system is the dominant consideration.

2. A Short Digression

In attempting to reproduce the sensation of flight, it is helpful to have an understanding of the physiological factors which are involved: the properties of the eye and vision, of the ear and balance, and of the brain and perception. There is no shortage of literature on these topics; the difficulty is to relate the complex medical and psychological descriptions into engineering terms.

Reference 1, which emerged from an AGARD Working Group, relates the properties of light and vision to the requirements for visual systems for simulators. Although it is made apparent that technology falls well short of providing a good reproduction of the world as perceived from an aircraft, trade-offs are available which produce an acceptable standard. The most obvious examples are the trade-offs between resolution, field of view, brightness and scene detail. Reference 2 identifies common visual effects, such as the illusions of distance, produced by size, perspective and shading.

Reference 3 deals with human balance mechanisms, in a similar way. The static dynamic properties of the semi-circular canals and otoliths are described in engineering terms, and their functions are respectively related to sensing angular and translational movements. Common illusions are cited, such as the sense of continued movement even when the stimulus is removed, in the absence of visual information.

A further class of illusion is seen when interplay occurs between motion and visual perception. The obvious example is vection, the illusion of movement which is seen by the passenger in a stationary train, when an adjacent train moves away. A similar effect, occurs to a subject at the centre of a rotating screen; a sensation of self-rotation occurs. Moving images on a fixed screen are equally effective in producing the illusion. Visually induced self motion is the basis for most simulator visual systems, including those employing a dome for wide angle displays.

Reference 4 describes this phenomonen in detail. Of special importance to simulator designers are the following features:

i) it is the peripheral field, rather than the central visual field, which must be stimulated,

ii) fixed objects in the foreground, such as a windscreen or instrument panel, do not affect the illusion, but fixed objects in the background, such as blemishes on the projection screen, can inhibit visually induced motion.

iii) there is an onset delay before the effect appear. The delay depends on rate of motion onset, and is reduced if the subject receives a washed-out motion stimulus in a direction to re-inforce the effect.

The second example of interplay between visual and motion cues is sometimes referred to as the 'Oculogravic Illusion'. Pilots of high performance jet aircraft are warned of the danger associated with forward acceleration after take-off in poor visibility. Without an attitude reference the combination of the forward acceleration and the gravity vector gives an impression of climbing flight. Nosedown correction then results in an unwanted rate of descent. The sensation is so powerful as to make pilots doubt the accuracy of head down intruments.

The same phenomonen is exploited in simulators with a visual system and six axis motion, to produce the sensation of sustained longitudinal and lateral forces. The initial acceleration is achieved by washed-out translation of the cockpit, and the steady state acceleration comes from an attitude change, with a suitable lag. The effect is particularly impressive during the take off and landing ground roll. Critical to this illusion is the correct presentation of the aircraft's attitude.

Still on the topic of simulating movement, most simulator users recognise the need to improve the pictorial quality of displayed scenes by texturing of surfaces such as fields, hills, and the runway. The underlying theories of perception are complex: cause and effect are difficult to separate (reference 5) Even so, the impression of most pilots is that texture enhances the height cues, and gives a much better impression of speed and direction over the ground, for landing and low flying tasks. Similarly, the sensation of turning, essential for air combat simulation, is enhanced by textured ground and sky images, projected inside a dome.

The other vital aspect of simulating visual cues is to provide spatial orientation. A distinctive horizon, over as wide a field of view as possible, is undoubtedly effective for this purpose. In recent years the concept of a two mode visual system has been suggested as a model (reference 6). The two modes are:

i) a focal mode, which employs the foveal, high resolution capability of the eye, to recognise objects, to read instruments, to aim guns, and so on. It answers the question "What are we seeing ?".

ii) an ambient mode, using peripheral vision with low resolution and wide field of view, to determine the quality of the surround. It answers the question "Where are we in space ?".

Use of the focal mode is a conscious activity, whereas use of the ambient mode is a subconscious activity. It is this latter mode that contributes, together with vestibular and somatosensory cues, to orientation and balance. One example to support this hypothesis is our ability to walk and read a book at the same time. In flying, the modes are clearly not divorced. If accuracte control of attitude or flight path is required, the focal mode prevails. In other circumstances, orientation is maintained through the ambient mode.

Although a wide horizon contributes to the sensing of attitude through the ambient mode (and this is the principle on which the Malcolm horizon indicator is based), the sensation is considerably re-inforced by other details such as clouds and ground. To distinguish such detail from the textured surfaces referred to earlier, the term spatial texture is useful.

The process of perception is extremely complex and interactive. The manner in which the eye and brain perceive colour, and adjust for changing circumstances, is remarkable. Similarly the control and stabilisation of eye movements in the body is, on engineering grounds alone, elegant and impressive, involving vestibular, proprioceptive and visual feedback. The common causes of disorientation are conflict of information received by the brain, leading to mis-interpretation. A more subtle form of disorientation occurs when there is a total lack of information - for example, in Arctic 'white-out' conditions.

The 'white-out' example illustrates that it is vital for a pilot to have orientation cues. Spatial texture plays a large part, particularly if the horizon is not well defined (which is usually the case), or if it is not in view. At night, stars, or lights on the ground are sufficient; in daytime, the sun, clouds, and objects on the ground fulfil the same role.

For short-term stabilisation, the objects do not need to be stationary; only some knowledge of their behaviour is needed. Examples are birds in flight, other aircraft, smoke, and precipitation. Flight in cloud can be aided if there is textural quality - for example, graded illumination. The important feature of spatial texture is that short-term changes of orientation can be detected, perhaps by reference to the windscreen arch or coaming.

If spatial texture is so important, it is equally important that the visual system used in flight simulators does not give false information of this type. Night/dusk systems, using a beam-penetration tube, are less likely to do this than daylight systems which rely on a TV raster scan to present the visual scene. If the raster is visible, and remains locked to the airframe, a false cue will result. The consequence of this false cue depends on how compelling is the information which is displayed correctly.

Now consider the presentation of orientation inside the dome of an Air Combat Simulator. One method of presentation is the sky/ground projector, using a point light source and a coloured transparency, mechanically driven in the three axes of rotation. Provided that the servos are responsive, no false orientation cue is present.

Another method is to use a fish eye lens to project a TV image over a wide field of view. The advantage is that height cues and movement over the ground are portrayed also. The disadvantage is that the resolution is poor, and the TV raster, which gives false spatial texture, is apparent. A point of note is that the new wide angle visual systems which employ an eye-slaved high resolution insert are meant to solve the problem of low resolution in the focal mode. What about the ambient mode?

3. A new configuration

Like other manufacturers of advanced military aircraft, British Aerospace at Warton makes extensive use of research simulators for development and clearance work. Considerable use has been made of a twin dome air combat simulator, incorporating projectors for target images and sky/ground images. Other simulators, using a three window CGI visual system, viewed through conventional beam-splitter optics, are available for the simulation of take off, low level flight, and landing.

None of these simulators currently provide motion cues.

A new simulator has now been commissioned which is intended to allow full-mission simulation, to support the design of the next generation of fighter aircraft. In this simulator, a departure has been made from the conventional method of introducing motion cues. Established practice, both in civil and military simulators, is to mount the cockpit and visual system on the motion platform. Civil aircraft simulators often use either four channels of beam-splitter optics and monitors, or systems similar to Rediffusion's WIDE. In either case, the result is a large and bulky load for the motion platform. Military training simulators including those for helicopters, assume similar proportions. More recently, simulators with even larger field of view requirements have resulted in the motion system carrying a dome and associated projection equipment. Naturally, the

performance requirements on the motion system increase in proportion to these loads, and the building to house the simulator needs to be large.

The new layout, described in reference 7, makes a radical departure, by installing a small six axis motion system inside a 30 feet diameter dome, and carrying on the motion system only the cockpit. All display devices are fixed, and are carried by a gantry behind the motion system (Figure 1). The display devices are

i) ground image projectors,
ii) sky/ground projector,
iii) air target projectors.

The ground image projectors are three low cost TV projectors (Electrohome ECP 2000) providing a field of view at the pilot's eye of 36° x 144°, with a brightness of 0.8 - 1 foot lambert, and a resolution of approximately 6 arc minutes/line pair. The sky/ground projector is an opto-mechanical device, using point-light sources shining through coloured transparent hemispheres, to produce images of sky and textured ground. The air target projectors (not yet installed) are similar to those used in the Air Combat Simulator. They project a monochrome high resolution image of an opponent aircraft, which can be positioned anywhere on the surface of the dome. Room is available on the gantry for additional display devices, such as additional ground image projectors, and missile flare projectors.

Even when the pilot's eye point is stationary, care is needed in the positioning of the display devices to achieve a satisfactory compromise between the conflicting requirements of field of view, distortion correction, clearance, and accessibility. In this layout, pilot's eye datum is located 1 m ahead and 1 m below the centre of the dome. Extensive use was made of computer graphics to predict shadowing, and the implication of off axis viewing on image correction. Critical in such analyses are the performance limits imposed by the choice of equipment. For example, the Electrohome projectors only have distortion correction in the vertical plane, which necessitates a symmetrical location to allow them to shine through the centre of the dome (Figure 2). One need which arises with this configuration is to match the horizon position and sky brightness of the CGI ground projectors with the sky/horizon from the sky/ground projector. One solution is to generate a black sky in the CGI system, and use only the sky/ground projector horizon to provide attitude reference.

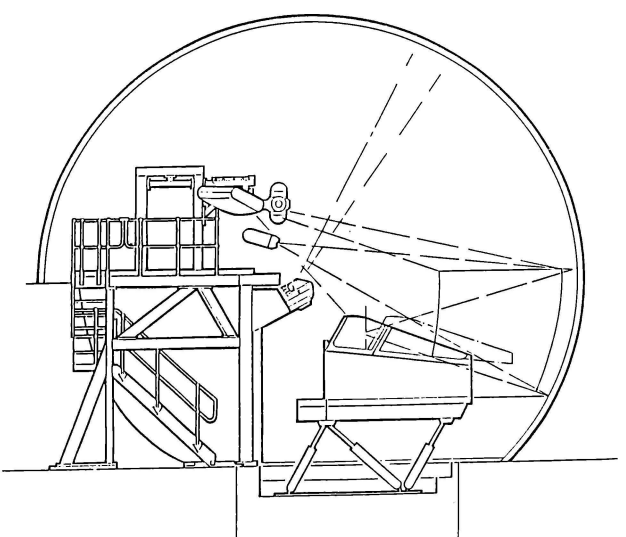


Figure 1. Layout - Side Elevation

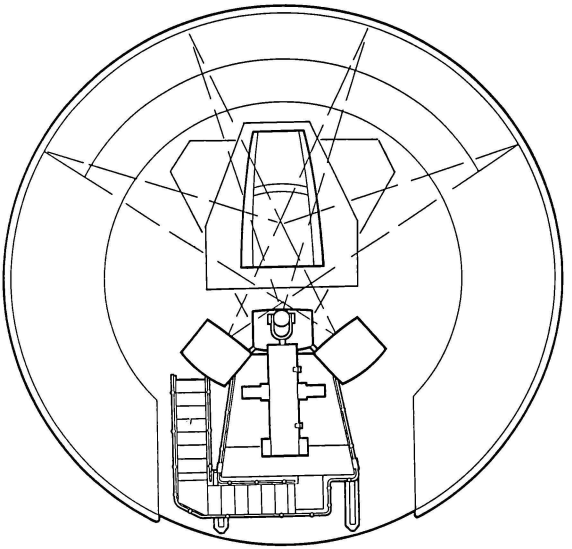


Figure 2. Layout - Plan

The use of a small synergistic motion system (McFadden 611B) is justified by considering the type of aircraft which the simulator will represent, and the purpose of the motion system. The type of aircraft, fighter/ground attack, is small and responsive compared to the large transport aircraft simulators which use 60 inch stroke motion systems. Consequently, the range of frequencies required is higher, and a smaller amplitude motion system (25 inch stroke) can be used. The purpose of the motion system also needs to be defined. The intention is to provide subjective realism, rather than motion fidelity. The primary function of the translational modes is to simulate vibration - aerodynamic buffet, engine vibration, undercarriage jolts, and runway contact. The rotational modes provide initial rotational accelerations, and steady state effects such as drift, or climb attitude. It should be noted that for the same geometry of base and platform attachment points, the maximum angular travels are independent of the scale of the motion platform.

Figure 3 shows the front elevation of the layout.

4. Advantages of the Layout

In a layout such as this, the dimensions of all equipments and their relative locations are critical. This is particularly true using a 9.1 m diameter dome as the display surface. Clearly, if a larger dome is used, problems of mechanical clearance, shadowing, and off axis compensation are eased. A motion system with greater travel might even be justified. To optimise a layout of this type, many iterations of configuration are needed. These can only be achieved by the use of computer graphics, which allow quick review of the layout from different aspects, and provides inspection of critical dimensions.

A further consideration is the need to introduce corrections to the projected visual displays for the movements of the cockpit. The corrections need to be based on the orientation and position of the pilot's eye-point. The only sure way to obtain this information is to measure the extension of each leg of the motion system, and to compute the mathematical transform which provides

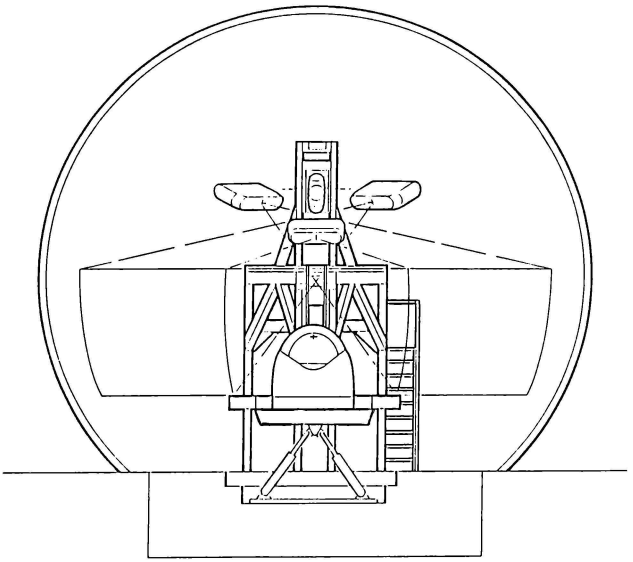


Figure 3. Layout - Front Elevation

three angles and three orthogonal displacements (Reference 8). The three angles so obtained are simply subtracted from the roll, pitch and yaw angles fed to the display. Corrections for displacements are made in a similar fashion. The displacement corrections are less critical than the angular corrections, and approximations are acceptable.

Having defined the configuration and applied corrective terms, the following benefits ensue.

i) Motion System Response. The load carried by the platform is reduced to a minimum, since it only carries the cockpit and crew. As a consequence, the performance of the motion system is improved, and wear and tear is lessened. A further advantage is the low centre of gravity of the supported mass, and its location close to the platform geometric centre. (Comparison is needed with the situation where three-window collimated optics and monitors are mounted on the front of the cockpit.)

ii) Visual Display Components. Off-the-shelf projectors may be used, since they are not subject to movement or vibrations. Freedom exists to change the type of projector without other engineering changes. More channels can be added, and the motion system performance is unaffected.

iii) Visual Display Dynamics. The initial accelerations seen (and felt) by the pilot come from the motion system. The resulting displacements are subtracted from the visual display drive signals, and initially the visual system does not respond. Thus the dynamic performance requirement for the visual system is reduced, giving more tolerance to lags/time delays in the image generator.

To illustrate this point, take a simple case of a first-order wash out motion drive law in bank, to the cockpit

$$\phi_c = \frac{t_1 s}{1+t_1 s}\ \phi$$

The displayed bank angle,

$$\phi_D = \phi - \phi_c = \frac{1}{1+t_1 s}\,\phi$$

Now suppose that there is an equivalent first order lag, t_2, between the visual input, ϕ_I, and the output, ϕ_D, of the image generator.

$$\phi_D = \frac{1}{1+t_2 s}\,\phi_I$$

$$\phi_I = \frac{1+t_2 s}{1+t_1 s}\,\phi$$

If $t_1 > t_2$, compensation for the lag t_2 is possible. In practice, the implementation is more complex, but the principle is clear. If the excursions in roll and pitch are small (for example, in ground roll), the extreme case occurs, in which all the perceived displacements can be produced by the motion system, and no inputs to the visual systems are necessary ($t_1 = \infty$).

iv) Realism. When simulating the ground roll, the motion system produces almost true motion fidelity, with the exception of forward acceleration. The visual system provides the effect of forward speed. Similarly, in flight at low speed, the trim attitude of the aircraft is accurately reproduced by the platform; transient and steady rate manoeuvres are reproduced by the visual system. The downward field of view benefits, compared to current systems, and on initiating landing flare, the cockpit moves, and the horizon and ground does not. (Figure 4)

During a crosswind approach, or in a navigational task with a crosswind, the drift angle, or crab angle, is attained by the motion system, and the runway, or track, is in the central part of the display. Kicking off drift causes the cockpit to move realistically onto the desired track.

v) Convenience. High on the list of requirements for both research and training simulators is for the cockpit to be 'as aircraft'. Preferably, entry and exit should be as on the aircraft, and normal flying clothing is preferred. These requirements are met by this configuration. Nor is there a need for pilot calibration of display optics.

Additionally, monitoring of performance is easy. The advent of head or eye slaved visual systems means that the casual observer will receive a false impression of the pilot's viewpoint; no such problems arise.

Visitors, a common feature of simulator facilities, can observe events without too much inconvenience to themselves or to the staff.

5. Operating Experience

Three types of aircraft have been simulated to date; an advanced fighter, a jet VSTOL, and an initial trainer. Although there are differences in the detailed motion/visual requirements for each of these types (and some tuning still remains to be done on each application), the subjective assessments have confirmed the benefits listed in section 3.

The increased realism is certainly there. The outside world, is without question outside the cockpit, and it is definitely the aircraft that moves initially, rather than the horizon. On this topic, it seems unlikely that we will ever see the situation which occurred all too frequently on fixed base simulators, that pilots new to simulators applied stick as if they were flying the horizon.

The initial response relies entirely on how well the motion system performs. In this case the McFadden 611B is an excellent

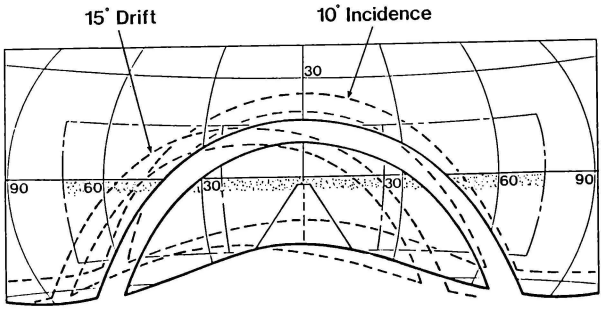


Figure 4. Field of View

choice - fast response, smooth, and quiet (Table 1). To a step input, it reaches 90% amplitude in less than 100 ms.

5.1 Corrections

The correction terms from motion to visual have been successfully mechanised. The task was helped considerably by the use of a graphic work-station. A real-time image of the 6 axis motion system was drawn on the display. The platform position was determined by three displacement signals, and three rotational signals. A second platform was then drawn, in a different colour, driven by the six leg displacements. The second platform image represents the true platform position, and has been used to visualise platform motions in the development of drive laws. The platform image determined by translational and rotational signals was driven by the outputs of the correction algorithm. By displaying simultaneously the two platform images, static and dynamic errors in the algorithm implementation were readily seen.

The acid test of successful correction was the installation in the cockpit of a head-up display. This display is fixed to the cockpit, so that its drive signals do not require correction terms. The HUD symbols show excellent correspondence with the projected outside world display - better in fact, than in fixed base simulators inside a dome, where lags or time delays associated with the projected displays are noticed.

5.2 Drive Laws

Over many years, effort and expense have been devoted to finding suitable drive laws for motion systems, both in the USA and Europe. The effort has been directed to three areas - hardware requirements, drive law algorithms, and human perception of motion.

The wealth of published data relates to motion systems which either carry the visual system with the cockpit, or do not include an outside world visual display. It was from such sources that our initial drive law implementation was taken. Reference 9 discusses this type of implementation, and the key parameters for fidelity. It shows that the coupling of translational and rotational axes is used universally to produce the sensation of steady acceleration. One difficulty which is discussed is that if only small translational travel is available, the required wash-out contravenes phase requirements, and the static gain must be reduced.

Having implemented these drive laws, on a platform with small travels in translation, we found that we did not have to address the problem. It was immediately apparent that with the new visual/motion configuration, forward acceleration and sideforce cannot be simulated by tilting the cockpit. The explanation is simple. Angular orientation cues inside the dome are powerful, and are not fixed to the cockpit frame of reference. The cues derive not only from the visual display but also from the screen edges, and such objects in the dome which are inadvertently seen by the pilot. Initially, they are space stabilised, and result in compelling space orientation. The simulation so far has used a projected three-channel display; the sky/ground projector is to be added later. Crude attempts to restrict the field of view only to this display suggests that the effect is unlikely to disappear with larger fields of view. Artefacts in the display itself - raster, noise, illumination - will contribute to spatial awareness.

The disappointment of not having this trick available to represent linear accelerations is more than compensated for by the realism of the orientation in roll, pitch and yaw. The primary function of the translational modes, to simulate high frequency effects, is not affected.

Table 1

McFadden 611B Motion System Performance

7000 lb payload

	x (min)	$\dot{x}$	$\ddot{x}$	$\dddot{x}$
Roll	±19°	±40°/sec	±250°/sec^2	±500°/sec^3
Pitch	+20°-18°	±40°/sec	±250°/sec^2	±500°/sec^3
Yaw	±23°	±40°/sec	±250°/sec^2	±500°/sec^3
Vertical	+14"-13"	±24"/sec	±1g	±5g/sec
Longitudinal	+21"-16"	±24"/sec	±1g	±5g/sec
Lateral	±18"	±24"/sec	±1g	±5g/sec

5.3 New Drive Laws

Implementation and proving of new drive laws is a painstaking task. Drive laws which suit one set of circumstances, for example, take-off and landing will not necessarily work at other flight conditions. Moreover, simulating a highly manoeuvreable military aircraft means that both small perturbation and maximum rate manoeuvres must be embraced by the laws. Nevertheless, progress has been made, and will be discussed by degree of freedom, rather than by application.

Pitch To simulate conventional aircraft, the most promising signal to the pitch channel is angle of incidence, unfiltered and unity gain. With this drive, initial pitching acceleration is well represented, and releasing the stick after a prolonged manoeuvre does not produce adverse effects. Target tracking is realistic. On take off, rotation occurs as on the aircraft, and the cockpit rotates realistically during landing flare. Such a drive signal needs more upward travel than down; our cockpit is rigged with a 5° bias.

To the pitch signal is added flight path angle, γ. The combined signal, $\alpha+\gamma$, is soft limited. The flight path signal gives a true sensation of the gravity vector in climbs and dives, within ±10° approximately. For wings-level flight, it follows that the pitch drive is θ, and no movement of the horizon relative to the screen is seen in gentle manoeuvres.

Yaw The complementary signal to incidence in pitch is sideslip, β, to drive the motion system in yaw. Initial yawing acceleration, due either to control excitation or turbulence is well represented, because unity gain, and no filtering can be used. In addition to β, the motion system is driven by drift angle, the difference between track and heading. As a result, the flight path direction is always down the middle of the visual display.

Roll Because sideforce terms are excluded, there seems to be little need for cockpit roll angles beyond about 15°. This is in contrast to the pitch and yaw channels, where large excursions are welcome. Satisfactory laws, based on washed-out roll rate, have been evolved for small perturbation manoeuvres, but rapid rolling is difficult to reproduce, and work continues. The initial response to controls can be matched subjectively, but roll reversal is then unsatifactory. A good test is to attempt a four point hesitation roll.

Translations With such limited travel, subjective realism must be the limit of our ambitions. The effects of wing buffet, turbulence, and ground contact have all been introduced. The current method is to drive directly from noise generator (suitably shaped), rather than to use signals from the full aircraft model. In this way, the full capability of the motion system can be enjoyed, and much more freedom of expression is available to the engineer evolving the drive laws. Later, it is our intention to use a form aircraft structural mode representation, to tie these arbitrary (though realistic) drives to engineering realities.

VSTOL The above comments apply to both the advanced fighter, and the initial trainer, in flight and on the ground. For a VSTOL aircraft in hover, incidence and sideslip are replaced as drive signals by pitch and yaw. The roll drive is bank angle with long term wash out. All signals are one to one gain. As speed increases, these signals are blended into those used for airborne flight. Although experience to date has been limited to representing a linear jet VSTOL model, either with good damping about all axes, or with very low damping about all axes, the results are encouraging. There is every indication that this configuration of visual/motion systems will produce VSTOL simulations of outstanding quality. In particular, it avoids confusion in the visual scene between attitude and height changes (and between heading and translation) which has marred earlier simulations. This benefit stems directly from the strong spatial orientation cue referred to earlier, and the fact that the cockpit movement shows attitude changes, and the visual display shows positional changes.

5.4 Pilot Comments

Pilot reaction to this configuration has been very favourable - there is an immediate affinity. Their evaluations have so far been confined to the initial trainer and the advanced fighter. In any research simulator, there are compromises on cockpit fit and other artefacts - critical comments have generally addressed such issues - 'you need a better noise system', 'the engine response was unrepresentative', ' I would have expected more yaw coupling in this type of aircraft'. The other difficulty of subjective assessment is that if you identify a possible deficiency before the assessment, it will become one. We have avoided this trap; nor have we begged for compliments.

The assessments have been coloured by the lack of accuracy and detail in the visual scene. Currently, we project a Tector Opdis display, a low-cost, random field pattern display with a non-textured runway [it is due for replacement soon by a modern three channel CGI visual system]. Close to the ground the Opdis display does not have the accuracy to perform precise landings. Even so, it has served our purpose well in demonstrating the principles. The field of view, 144° x 36°, is sufficient for take-off, landing, and stately manoeuvres, but for vigorous manoeuvres, it would be preferably to increase the vertical field of view in an upward direction, to remove any intrusion from the top edge of the screen. Current plans see an

additional 9° as sufficient for the initial trainer. The advanced fighter case will be covered when the displays for air to air combat are added.

One pleasing aspect is the promise that this configuration gives of more realistic landing flare simulation. There are signs that pilots 'feel' for the ground more with this configuration than with others, during landing flare. Of course, more of the ground is in view. When the more accurate visual system is installed, quantitative measurements of landing performance will be made.

The overall general impression, is that the claim for greater realism is justified. Evaluations of flying qualities, and of weapon aiming, should be easier to make in this simulator. We are able to do a direct comparison with a fixed cockpit simulator which has the Opdis display presented by Barco monitors through beam splitter optics. One pilot made the strange comment that he was 'closer to the outside world' in that cockpit, than the cockpit in the dome.

6. Conclusions

6.1 A new approach to combining visual and motion cues in a flight simulator has been described, in which the display elements remain fixed, and the cockpit moves. The key to the use of this method is in the corrections to the visual display for movements of the cockpit.

6.2 The approach has advantages from considerations of hardware, and of operating convenience.

6.3 Traditional methods of motion drive mechanisation have been discarded. The orientation cues due to spatial texture no longer permit the oculogravic illusion to be exploited. The benefit is very good spatial orientation for the pilot, and a sense of being 'in flight'.

6.4 Other methods of representing linear accelerations need to be added. These include a 'g' seat, a 'g' suit, and dimming of the visual display with g.

6.5 The configuration could be applied to most types of aircraft and helicopter simulators. Each application would need careful consideration of the visual/motion implementation.

6.6 There is a lesson also for designers of visual systems in general. If the display reference system is fixed to the cockpit or to the pilot's head, display imperfections such as TV raster lines, or poor edge matching may be intrusive. They will contribute to false spatial awareness, which could be compensated by more pictorial content. Or the pilots will learn to live with it, as they have in the past.

References

1. AGARD FMP Working Group "Characteristics of Flight Simulator Visual Systems". AGARD AR 164 May 1981

2. Gullen, Robert K., et al "The Computer Image Generation - Applications Study". AFWAL-TR-80-3075 July 1980

3. Peters, R.A. "Dynamics of the Vestibular System and their relationship to Motion Perception, Spatial Disorientation, and Illusions". NASA CR 628 April 1969

4. Young, L.R. "Visually Induced Motion in Flight Simulation" AGARD Conference on "Piloted Aircraft Environment Simulation Techniques". AGARD CP 249 October 1978

5. Chapplow J.W., Smart J.A. "Putting Texture in Perspective". RAeS Conference on Flight Simulation - Avionic Systems and Aeromedical Aspects. April 1982

6. McNaughton, Col Grant B., "Background paper on the role of visual system in spatial orientation/disorientation" Conference on Peripheral Vision Horizon Display. NASA CP 2306 1984

7. Barnes, A.G. "The integration of a six axis motion system and a wide angle visual system inside a dome". Paper presented at Royal Aeronautical Society Conference on Advances in Visual and Motion Systems. London April 1986

8. Dieudonne, J.E. et. al. "An actuator extension transformation for a motion simulation, and an inverse transformation using the Newton-Ruphson method". NASA TN D-7067 November 1972

9. Tomlinson, B.N. "Simulation Motion Characteristics and Perceptual Fidelity, a Progress Report". AGARD CP 408 October 1985

87-2498

A NEW SIMULATION MODEL BUILDING PROCESS FOR USE IN DYNAMIC SYSTEMS INTEGRATION RESEARCH

P. Douglas Arbuckle*, Carey S. Buttrill*
NASA Langley Research Center, Hampton, Virginia
and
Thomas A. Zeiler†
PRC Kentron, Inc., Hampton, Virginia

Abstract

A framework to build simulation models for aircraft dynamic systems integration is described. The objective of the framework is increased simulation model fidelity and reduced time required to develop and modify these models. The equations of motion for an elastic aircraft and their impact on the framework are discussed in broad terms. A software tool which automatically generates FORTRAN routines for tabular data lookups, the language used to develop a simulation model, and the structures for passing information into a simulation are discussed. A simulation variable nomenclature is presented. The framework has been applied to build an open-loop F/A-18 simulation model. This example model is used to illustrate model reduction issues. Current deficiencies in the framework are identified as areas for future research.

Nomenclature

Symbols

C^{as} second-order momentum coupling matrix, for n antisymmetric elastic modes, (3 x n);
$= \left[\underline{h}^{as}_{j1}\eta^j : \ldots : \underline{h}^{as}_{jn}\eta^j \right]$

C^{sy} second-order momentum coupling matrix, for n symmetric elastic modes, (3 x n)

$\underline{F}$ vector of total applied force on the aircraft, (3 x 1)

g_{11} gravity gradient with respect to altitude

g_o gravitational acceleration at sea level

$\underline{h}_{jk}$ vector of residual mass coupling between angular and elastic momentum, (3 x 1); $\underline{h}_{jk} = -\underline{h}_{jk}$

H altitude above mean sea level, ft

H_p body-frame origin altitude above local Earth frame

$[J]$ total inertia matrix of the aircraft in its deformed state, (3 x 3);
$= [J_o] + [\Delta J]_j \eta^j + \frac{1}{2}[\Delta^2 J]_{jk}\eta^j\eta^k$

$[J_o]$ standard inertia matrix of the aircraft in its unloaded reference condition, (3 x 3);
$= \begin{bmatrix} I_{xx} & 0 & -I_{xz} \\ 0 & I_{yy} & 0 \\ -I_{xz} & 0 & I_{zz} \end{bmatrix}$

$[\Delta J]_j$ first partial derivative of the aircraft inertia matrix $[J]$ with respect to the j^{th} mode, (3 x 3);
$[\Delta J]_j = [\Delta J]^T_j = \frac{\partial}{\partial \eta_j}[J]$

$[\Delta^2 J]_{jk}$ second partial derivative of the aircraft inertia matrix $[J]$ with respect to the j^{th} and k^{th} modes, (3 x 3);
$[\Delta^2 J]_{jk} = [\Delta^2 J]^T_{jk} = [\Delta^2 J]_{kj} = \frac{\partial^2}{\partial \eta_j \partial \eta_k}[J]$

l_Z vector which is the third column of the Earth-to-body-frame direction cosine matrix, (3 x 1);
$= [\ -\sin\theta \quad \sin\phi\cos\theta \quad \cos\phi\cos\theta\ ]^T$

$\underline{L}$ vector of total applied moment on the aircraft, (3 x 1)

m total aircraft mass

M Mach number

M_{aug} "augmented mass" matrix, see fig. 2 for definition

M_{jk} mass coupling between the j^{th} and k^{th} modes

Q_{η_j} generalized force on the j^{th} elastic mode

$\underline{V}$ velocity vector of the body-frame origin with respect to the inertial frame, (3 x 1)

α angle-of-attack, deg

β angle-of-sideslip, deg

η_j generalized coordinate corresponding to the j^{th} elastic mode

η^j, η^k elastic mode generalized coordinates in the context of an indicial summation over j (or k) from 1 to n ; where n is the number of elastic modes

θ Euler pitch angle

ϕ Euler bank angle

$\underline{\omega}$ rotational velocity vector of the body frame with respect to the inertial frame, (3 x 1)

Operators

$\{\ \dot{}\ \}$ time rate of change of $\{\ \}$ with respect to the Earth frame

$\{\ \ddot{}\ \}$ second derivative of $\{\ \}$ with respect to time, relative to the Earth frame

$\{\ ^{\circ}\ \}$ time rate of change of $\{\ \}$ with respect to the body frame

$\{\ \}^T$ transpose of $\{\ \}$

---

* Research Engineer, Member AIAA
† Project Structures Engineer, Member AIAA

Abbreviations

i.c. initial condition
n.d. nondimensional

Acronyms

ACSL	Advanced Continuous Simulation Language
CFD	Computational Fluid Dynamics
DATCOM	Data Compendium
EAL	Engineering Analysis Language
FIT	Functional Integration Technology
ISAC	Interaction of Structures, Aerodynamics and Controls
HARV	High Angle-of-Attack Research Vehicle
LaRC	Langley Research Center
MCAIR	McDonnell Aircraft Company, McDonnell Douglas Corporation
NASTRAN	NASA Structural Analysis
RFA	Rational Function Approximation
RTS	Real-Time Simulation

Introduction

The design of future aircraft will require multidisciplinary integrated design and analysis. Agility requirements for future fighters are such that unsteady aerodynamic effects (dynamic stall, etc.) may one day become more important than classical static performance criteria.[1] Experimental forward-swept-wing configurations have demonstrated significant coupling of rigid-body pitch rate and the wing first bending elastic mode,[2] and such configurations have been proposed for future fighter designs. Flight test programs have demonstrated the feasibility of improving aircraft performance by using feedback control to provide static stability, maneuver load alleviation, and/or increased flutter mode damping.[3] Current research and emphasis on supermaneuverable fighters with capabilities such as "point and shoot" and post-stall maneuvering using all-axis thrust-vectored control[4] suggests that aircraft angular response rates may one day be limited only by concern for the pilot's ability to function. These facts and trends all point to the conclusion that future aircraft designs, particularly fighters, will be "integrated" in some fashion.

By definition, today's aircraft designs are "integrated" in a process that might be termed "Subsystem Integration." In this process, the aircraft subsystems are designed independently and then integrated in a manner which minimizes interactions. The trend in integration is moving toward "Functional Integration." In this case many functions would be considered early in the design process. To keep computations practical, several systems would be functionally integrated rather than attempt to simultaneously optimize all of the design variables. In the "Configuration Integration" process, traditional design constraints would be relaxed in a manner to achieve large performance gains through technology integration.[4]

Aircraft designed using "Functional" or "Configuration" integration techniques are uniquely characterized by (1) the extent to which the aircraft disciplines or subsystems are combined and (2) the use of embedded digital control systems to accomplish the integration. Often, interactions that in the past were regarded as undesirable are used in a controlled, beneficial manner, making the loss of the control system unacceptable and usually unsafe. The primary distinction of these designs from most current aircraft is the flight safety dependence of the integrations.[5]

Applications of aircraft dynamic systems analyses fall into three categories which are related to the three approaches to integration described above. These categories are: (1) applications to solve problems arising late in the aircraft development process, i.e., a "fix"; (2) applications to a frozen configuration to achieve some benefits; and (3) applications early in the design process that impact the configuration in a manner to greatly enhance performance.[4]

The motivation for this research arose from the need to develop and document a methodology for dynamic systems integration in aircraft configuration design. Some of the near-term technologies required to implement this methodology are being developed and applied by the Functional Integration Technology (FIT) team at Langley Research Center (LaRC).

Two ultimate objectives of the dynamic systems integration methodology are:

1) an order of magnitude improvement in the effectiveness of simulation in the design process, accomplished by increasing model fidelity while reducing the time required to develop and modify simulation models; and
2) removal of unfavorable dynamic systems characteristics while taking advantage of favorable system interactions to enhance performance, extend the flight envelope, and exploit new technologies at all design stages.

Achieving these objectives will allow flying qualities engineers and control system designers to play a first-order role in the aircraft configuration design process.

This paper describes the development of a framework to build simulation models which satisfy the first objective listed above. Design of this framework has been driven by at least four factors:

1) The desire to have a paperless interface with the Real-Time Simulation (RTS) facility at LaRC. This includes the ultimate capability to deliver validated aircraft control laws as FORTRAN subroutines to be inserted into the real-time simulation code.
2) The equations of motion for an elastic aircraft.
3) The use of the Advanced Continuous Simulation Language (ACSL)[6] software to develop aircraft simulation models.
4) The current technology level of analysis and design tools, both in terms of what quantities can be computed for use in the simulation equations of motion and in terms of what quantities are required as inputs to control law algorithms.

Also, several philosophies developed by Radovcich[7] have guided the framework development:

1) The computer methodology developed must be flexible and highly modular to deter obsolescence as new engineering analysis tools become available.
2) Each discipline will, in general, define its own modeling requirements. Modeling tasks which involve mul-

tiple disciplines will be integrated by a designated discipline.

The framework has been exercised to build a simulation model of the F/A-18 aircraft. The current simulation model is an open-loop model of the continuous airplane plant. The flight control system of the F/A-18 has been used to solve various problems which surfaced during F/A-18 flight test development.[8] This fact made the inclusion of the control system model into the simulation a priority. However, since this F/A-18 simulation was to be used as an initial testbed for developing and refining dynamic systems integration techniques, it was critical that an open-loop model be developed for analysis by control law algorithm researchers. Therefore, the model of the flight control system will be included in the simulation at a later date.

This paper will also present a proposed system for assigning variable names in aircraft simulation models based on nomenclature developed by the authors.

Software Tool for Tabular Data Models

The RTS facility at LaRC uses a specific file format for storing tabular data values to be used by function lookup routines. This file format is defined as the input to a LaRC-developed software tool that aids in building FORTRAN code blocks to perform the table lookups. The software tool is called DATAFIX.

Typical applications for table lookup functions in aircraft simulation include: calculation of total aerodynamic coefficients; use of engine performance tables; and implementation of control laws with scheduled parameters.

The DATAFIX format was adopted for use in the simulation model building framework as a step toward achieving the desired paperless interface with the LaRC RTS facility. We have enhanced the DATAFIX software in several ways, including minor extensions to the input file format and addition of a "translator" feature. The enhanced software tool is named FITDFX. A schematic representation of the use of FITDFX in the simulation model building framework is shown in figure 1. Appendix A contains the specifications of the extended-DATAFIX file read by FITDFX.

The translator software first reads a file of tabular data values to determine the functional dependencies of the data and the ranges of the independent variable values. The functional dependencies are automatically written in one-line FORTRAN-like statements called "primitives." As an example, the tabular data for the F/A-18 aerodynamics model contains a table lookup for lift coefficient due to flap deflection as a function of angle-of-attack and Mach number, or

$$C_{L_{\delta_f}} = f(\alpha, M)$$

The primitive for this lookup is written as

CLDF = $FCLDF (ALFDG , MACH)

The "$F" string is the operator which identifies the primitives for the translator.

The engineer uses a text editor to manipulate and combine primitives, as well as to input standard FORTRAN statements that compute quantities for the simulation and provide internal documentation. FITDFX uses the file augmented by the engineer to automatically generate the necessary FORTRAN subroutines to load the tabular data and compute the lookups. For a typical steady aerodynamics package (step ⑥), up to 80 percent of the FORTRAN code lines may be generated without human intervention. At least 50 percent of the FORTRAN code lines are generated for typical engine models.

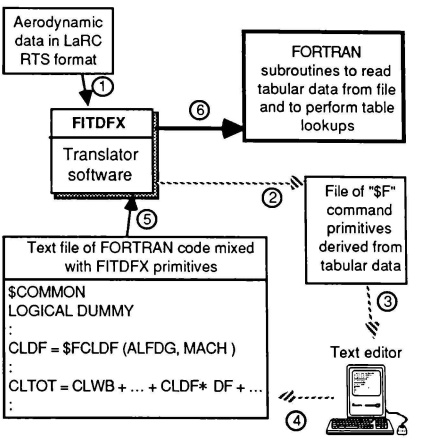


Fig. 1 Use of FITDFX software to build FORTRAN code blocks in simulation model building framework

Since the DATAFIX file format has been adopted as a "standard" for the simulation model framework, all tabular data will be converted via automated software tools to the extended-DATAFIX format (see Appendix A) as a condition for use in the framework. One such tool has been developed at LaRC to convert output from the USAF Digital DATCOM program to extended-DATAFIX format. Similar tools for converting wind tunnel data, output from computational fluid dynamics (CFD) codes, or an engine deck could easily be developed.

Elastic Aircraft Equations of Motion

Before a vehicle simulation can be developed, the equations of motion of the vehicle must be known. Further, the equations of motion must include all terms necessary to accurately simulate the desired dynamics.

The elastic aircraft equations of motion are examined using Lagrangian mechanics. The aircraft is idealized as a collection of lumped masses and lumped inertias being displaced about a non-inertial mean reference body-axis frame. The elastic aircraft equations of motion as developed are:[9]

Translational Momentum

$$m\dot{\underline{V}} = \underline{F} - m(\underline{\omega} \times \underline{V}) + m(g_o + g_H H_P) I_Z \qquad (1a)$$

Angular Momentum

$$[J]\dot{\underline{\omega}} + \underline{h}_{jk}\,\eta^j\,\ddot{\eta}^k = \underline{L} - \underline{\omega} \times [J]\,\underline{\omega} - [\dot{J}]\,\underline{\omega} - \underline{h}_{jk}\,\dot{\eta}^j\,\dot{\eta}^k - \underline{\omega} \times \underline{h}_{jk}\,\eta^j\,\dot{\eta}^k \quad (1b)$$

j^{th} Elastic Mode

$$M_{jk}\ddot{\eta}^k - \dot{\underline{\omega}}^T \underline{h}_{jk}\,\eta^k = Q_{\eta_j} - M_{jj}\omega^2_j\,\eta_j + 2\underline{\omega}^T \underline{h}_{jk}\,\dot{\eta}^k + \tfrac{1}{2}\underline{\omega}^T\{[\Delta J]_j + [\Delta^2 J]_{jk}\eta^k\}\,\underline{\omega} \quad (1c)$$

The notation η^k indicates a summation over k from 1 to n. Also, the notation $\underline{a}^T\underline{b}$ is equivalent to $\underline{a}\cdot\underline{b}$. The equations (1) are not new. The only difference from what is typically found in the literature for aircraft is that normally neglected nonlinear inertial coupling terms are retained. These terms involve products of rigid-body angular rates, structural deformations, and structural deformation rates. The terms are neglected in typical formulations by assuming that the body-axis rotational rates are small. The coupling terms arise from two sources.

The first source is the change in the total aircraft inertia matrix due to elastic deformation. If deformation is described as a linear combination of elastic modes with their time-dependent participation coefficients, the inertia matrix is subject to first and second partial derivatives with respect to the modes. The terms $[\Delta J]_j$ and $[\Delta^2 J]_{jk}$ are the result.

The second source of nonlinear inertial coupling arises from the fact that given a modal description of deformation, the mode shapes can only be forced to satisfy the first-order mean-axis conditions ("practical" mean-axis conditions[10]). The term $\underline{h}_{jk}$ represents a second-order coupling between angular and modal momentum and is calculated by integrating vector cross-products of the mode shapes over the aircraft.

The nonlinear inertial coupling terms make virtually no difference in linear stability derivatives calculated at straight and level flight conditions. It is conjectured that their impact on elastic aircraft response at high angular rates for a configuration with stores could be significant.[9]

The implemented form of the equations of motion in the simulation is for an aircraft which is symmetric about the body-frame x-z plane. The implemented equations take computational advantage of this symmetry by partitioning appropriate vectors and matrices into symmetric and anti-symmetric components. This partitioning permits the known zero elements in the equations to be arranged in predetermined submatrices which are not operated upon in the actual simulation code. Figure 2 shows how the partitioning is accomplished. The symbol Σ denotes a summation of the indicated vectors. $\underline{F}'$ represents the various force vectors due to aerodynamic loads, gravity and inertial effects. Similarly, $\underline{L}'$ represents the various moment vectors. $\underline{Q}$ represents the various contributions to the generalized force vectors affecting the elastic degrees of freedom.

Simulation Language

As implied in the Introduction, the Advanced Continuous Simulation Language (ACSL)[6] has been used in the simulation model building framework. Major factors in this decision were the commercial availability of ACSL and our possession of the software. Other reasons supporting the selection of ACSL include:

1) It can automatically generate a linear system quadruple from the full nonlinear set of differential equations.
2) It allows the creation of user-written macros and can call user-written FORTRAN subroutines.
3) It provides a rigorous blending of discrete and continuous dynamic systems.

ACSL does impose some limitations and programming rigor on its users. For example, due to the ACSL requirement that differential equations be in explicit form, the elastic aircraft equations of motion were written such that all acceleration-dependent force terms appear on the left side of the equations. The force terms, arising from unsteady aerodynamics, were brought into the mass matrix as "apparent mass" terms. The resulting time-dependent mass matrix was named the "augmented mass matrix" (fig. 2).

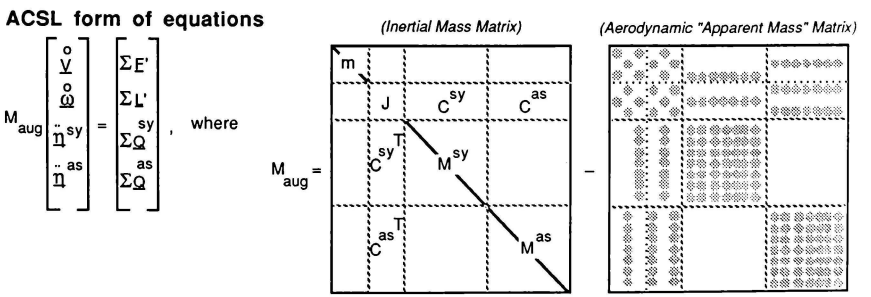


Fig. 2 Elastic symmetric aircraft equations of motion written with augmented mass matrix, M_{aug}

The time dependence of the augmented mass matrix, due to the apparent mass terms and the submatrices [J], C^{as} and C^{sy}, causes a potential matrix inversion each time the equations of motion are evaluated. The second-order Runge-Kutta integration algorithm used in the simulation requires two evaluations of the equations of motion per time step. Therefore, a possible requirement was presented of inverting a matrix of size (6+*n*)-by-(6+*n*) two times per integration time step. This matrix inversion would have imposed a significant computational burden on the simulation, so other options were explored.

It was observed that the diagonal elements of the augmented mass matrix were, for aircraft, always large relative to the other matrix elements. With this insight, a zero-pivoting Gaussian elimination algorithm was used to make the augmented mass matrix lower triangular and therefore solvable without matrix inversion. This approach assured a rigorous computation of the aircraft's motion (and its linear system quadruple) yet introduced no measurable computational penalty in the ACSL environment.

The philosophy adopted for use of ACSL in the simulation model building framework is that all dynamics which are physically continuous are modeled in ACSL. This includes actuator and sensor dynamics, engine dynamics, and the equations of motion. Physically discrete dynamics, such as a digital flight control system, are not coded in ACSL. These dynamics are coded in FORTRAN and accessed by the simulation through subroutine calls which occur at discrete time increments.

Data Requirements and Information Flow

The input quantities required by the simulation are defined by the equations of motion formulation and include normally computed quantities – geometry and mass data, nonlinear steady aerodynamic data, engine data – as well as quantities from the structural model (mode shapes, frequencies, etc.) and from the unsteady aerodynamics model (generalized forces).

Nonlinear Steady Aerodynamics Model

The steady aerodynamics model and its documentation were obtained from the LaRC RTS facility as extracted from their operational F/A-18 simulation. The RTS aerodynamics model consists of FORTRAN subroutines which access a tabular data base of aerodynamic coefficients. These subroutines can not be directly used in the simulation model building framework since they contain machine-dependent DATA statements and call assembly language routines.

The F/A-18 steady aerodynamics data base is defined within the following range of aerodynamic parameters:

$$-10° \le \alpha \le 90°$$
$$-20° \le \beta \le 20°$$
$$0.2 \le M \le 2.0$$
$$0\text{ ft} \le H \le 60{,}000\text{ ft}$$

The data base was delivered by the LaRC RTS facility in the DATAFIX file format. This file was converted to the extended-DATAFIX format and processed as described previously (see "Software Tools for Tabular Data Models"). RTS documentation of the steady flow aerodynamics model was used extensively during the editing process of combining and manipulating primitives. The FITDFX software was then used to generate the FORTRAN subroutines which load the tabular aerodynamic data into program memory and perform linear interpolation lookups as required by the simulation.

Engine Model

The F404 engine model and its documentation were obtained from the LaRC RTS F/A-18 simulation. The RTS engine model consists of FORTRAN subroutines which access a tabular data base of engine performance and dynamics parameters. These subroutines also contain machine-dependent DATA statements and call assembly language routines.

Both the throttle-commanded steady-state thrust level and the dynamic response characteristics of the engine model are based on the engine airflow rate as determined from a table lookup. Afterburner dynamics are switched in at a threshold based on the engine airflow and commanded thrust. A model of the thrust-vectoring system proposed for the NASA F/A-18 High Angle-of-Attack Research Vehicle (HARV) is also included and may be optionally activated.

The data base was delivered in DATAFIX format and processed in a manner largely analogous to the steady aerodynamics data base. The engine dynamics, being physically continuous, were modeled in ACSL, in keeping with the philosophy described previously (see "Simulation Language").

Structural Model

A NASTRAN beam half-model (fig. 3) of the F/A-18 was obtained from the McDonnell Aircraft Company, McDonnell Douglas Corporation (MCAIR), and translated into EAL (Engineering Analysis Language)[11] so that the data management and processing capabilities of EAL could be utilized. The structural model was analyzed to determine the free-free vibration modes with both symmetric and antisymmetric boundary conditions and to determine internal modal load coefficients.

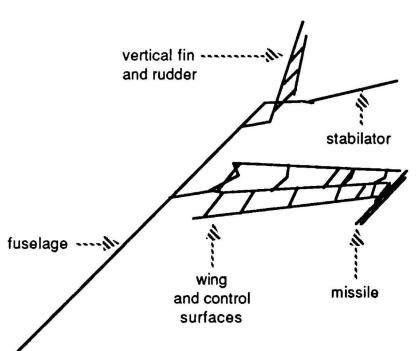


Fig. 3 Finite-element beam half-model of the F/A-18

With external FORTRAN programs, the mode shapes were analyzed to determine their satisfaction of the practical mean-axis conditions.[10] Though the free vibration eigenproblem was solved for free-free boundary conditions, the modes were found to be somewhat "dirty" in that the mean-axis conditions were not satisfied to machine accuracy level. Previous experience indicates that "dirty" mode shapes are expected whenever a computer solution to a sizeable eigenproblem (especially one with several zero eigenvalues) is attempted. Although their work deals with the different situation of using restrained structure characteristics in modeling a free vehicle, Rodden and Love[12] show the potential for noticeable effects if the mean-axis conditions are not satisfied.

Small translational and rotational corrections were computed and applied to the mode shapes. These corrections preserved the mode shapes and merely reoriented them in order to satisfy the practical mean-axis conditions. Therefore the load coefficients computed within EAL remained valid. Using these corrected mode shapes, the nonlinear inertial coupling quantities and generalized masses required by the equations of motion were computed and arranged on a data file named JDATA (see Appendix B) to be loaded into the simulation.

Unsteady Aerodynamics Model

The same mode shapes were used in the ISAC[13] programs to compute generalized unsteady aerodynamic loads using doublet-lattice theory for a range of reduced frequencies and Mach numbers. For each Mach number, a rational function approximation (RFA) for the transfer function of the unsteady aerodynamic loads was determined by a least-squares fit to a table of oscillatory loads at various reduced frequencies and for a selected set of aerodynamic "lags."[14] The lags can also be optimized, if desired, to minimize the least-squares error.[15] The s-plane fits are placed on a data file called SPDATA (see Appendix C) to be loaded into the simulation.

Structural Loads

Selected load coefficients were placed in another data file named LOADS (see Appendix D) to be loaded into the simulation so that time histories of internal structural loads could be computed and displayed for a set of predetermined structural stations. Alternatively, a file of modal coordinate (η_j) time histories from a simulation run can be made available to the structural analyst for more detailed and extensive loads analyses.

Information Flow

Figure 4 shows a flowchart of the basic components of the simulation model building framework as solid lines; the dotted lines indicate the links between the framework and other elements of the dynamic systems integration methodology. The heavy dotted lines indicate successfully exercised pathways to existing tools. The light dotted lines indicate pathways which are planned but have not yet been developed.

The starting points for building a simulation model using the framework are the EAL structural model, aerodynamic data, and engine data. The aerodynamic and engine

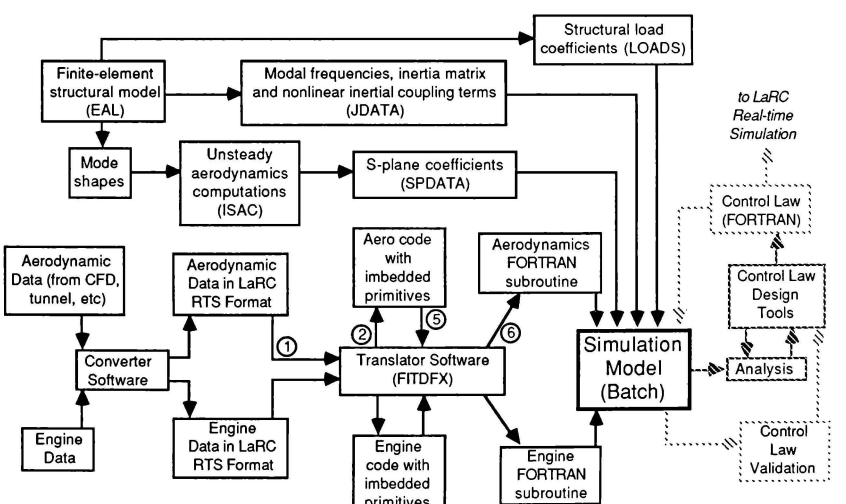


Fig. 4 Simulation Model Building Framework Flowchart

data are converted to the extended-DATAFIX file format (see Appendix A) and processed to generate the steady aerodynamics model (① - ⑯ correspond to fig. 1) and engine model as described. Simultaneously, the EAL model is exercised to generate mode shapes for input to the ISAC programs, as well as information for the JDATA and LOADS files. The ISAC programs compute the s-plane coefficients for the RFA formulation of the unsteady aerodynamics, forming the SPDATA file. Ideally, the various data files and FORTRAN subroutines to be used by the ACSL simulation would become available at approximately the same time. After verifying and validating the complete simulation model, it can be exercised to generate results as required.

Variable Nomenclature

Development of the simulation building framework, like development of a RTS model, has involved the efforts of several engineers and programmers and has included receipt of data, documentation, and several FORTRAN subroutines from "outside" sources (MCAIR and the LaRC RTS facility). This effort has given us a keen sense of the desirability of a "standard" variable nomenclature for coding simulation models. Without a "standard" variable nomenclature, it was generally impossible to follow the coded logic without having the documentation in hand.

The authors found this situation to be undesirable and developed a system, used by all personnel working on the project, for naming code variables. This should not be confused with issuing a variable dictionary for project personnel to use. The philosophy behind the variable nomenclature system was that any given variable name could be constructed from a set of predefined mnemonics (base names; prefix and suffix modifiers), given an appropriate set of construction rules. This philosophy is based upon ideas enunciated by Mitchell.[16] A set of mnemonics developed by the authors is listed in table I.

The construction rules for creating a variable name are as follows:

1) In keeping with the ANSI FORTRAN 77 standard and the requirements of ACSL, all variable names are six characters or less.
2) Prefix and suffix modifiers may only be used in combination with a base name.
3) "Special" names (see table I) may not have modifiers.
4) Multiple prefixes may only be used if their nesting levels are different. The highest nesting level prefix will be leftmost, followed by the base name.
5) Multiple suffixes may only be used if their nesting levels are different. The highest nesting level suffix will be rightmost.
6) Base name truncation may be used to meet the requirement that all names be six characters or less. If truncation is required, remove the right character of the base name.

As an example of using the nomenclature system and construction rules, consider a variable defining β in units of degrees at the aerodynamic reference point. The base name for β is BET, the suffixes are DG and RF, respectively. The suffix DG has a nesting level of two, and the suffix RF has a nesting level of three (see table I). Constructing the variable name using rule number five results in BETDGRF, which is a seven-character name and violates rule number one. Using rule number six, remove the right character of BET to give us BEDGRF as the variable name which satisfies the construction rules.

ACSL provides the capability for creating and maintaining a file-based simulation variable dictionary.[6] The implementation of the dictionary capability quickly identifies variables which have undefined values and definitions. The ACSL dictionary was installed as an option in the F/A-18 simulation and was used in the development of the variable nomenclature system. The dictionary file developed for use with the F/A-18 simulation is used as the permanent documentation record of all assigned variable names in the main simulation program.

Model Size and Model Reduction

One of the philosophies inherent in the development of dynamic systems integration methodology and the simulation model building framework is that the resulting large, high-fidelity simulation will be run in a batch mode and will be regarded as a "truth" model. Further, the RTS model as well as models for dynamics analysis and control law design, will be reduced models of the "truth" simulation.

The technical problem is how to reduce the "truth" simulation model to a size suitable for RTS, dynamics analysis, or control law design while maintaining the desired accuracy. It is probable that new techniques will be required to accomplish this reduction in a reasonably expedient way. Some insight into the amount of model reduction required is provided by considering the F/A-18 simulation developed with the model building framework.

The number of states in the open-loop F/A-18 simulation depends strongly on the number of elastic modes required to describe the structural dynamics and the number of aerodynamic lags required to describe the unsteady aerodynamics. For the original F/A-18 simulation implementation, the states are broken down as follows:

6	Rigid-body
5	Altitude/Quaternions
14	Engine (core plus afterburner)
3	Gusts
19	Control Surfaces (1st order actuators, 3 surfaces; 2nd order actuators, 8 surfaces)
40	Elastic Modes (20 modes)
104	Aerodynamic Lags (4-lag formulation)
191	TOTAL states

Reducing the number of aerodynamic lags by one in the unsteady aerodynamic formulation reduces the number of total model states by six plus the number of elastic modes (26 for the original F/A-18 simulation). Unfortunately, as the number of aerodynamic lags are reduced, obtaining an "accurate" RFA representation of the unsteady aerodynamics becomes more difficult and time consuming. Research has

been conducted to develop techniques to reduce the number of aerodynamic lags while preserving a given level of approximation error.[15] These techniques will be applied in the future as appropriate to compute unsteady aerodynamics in the simulation model.

The aerodynamic lag formulation poses a particular problem for implementing RTS approximate models of the "truth" simulation model. The "speed" of the lag states are determined by the range of reduced frequencies necessary to model the aircraft dynamics under investigation. The original implementation of the unsteady aerodynamic data base for the open-loop F/A-18 simulation is such that at M=0.9, sea level, the eigenvalue of the fastest aerodynamic lag state is a real pole at approximately –349 radians per second. To insure that this state does not become numerically unstable, the simulation uses a time step of 0.002 seconds, which is not achievable in most current RTS facilities. RTS studies will typically consider a lower range of modal frequencies and a reduced number of elastic degrees of freedom than used in the "truth" simulation. For such studies, it may be possible to choose the aerodynamic lag pole magnitudes such that the RTS time step can be increased to about 0.01 seconds.

Up to this time we have virtually ignored the CPU and memory requirements of simulations built using the framework. This was done primarily to insure that the methodology developed would be relatively independent of the computer hardware and software environment in which the framework would be exercised. Although relatively machine-independent, the open-loop F/A-18 simulation is usually exercised on a MicroVAX II minicomputer. In this environment, the simulation operates at approximately 500 times slower than real time and uses approximately 250,000 words of memory. The execution speed of the simulation depends strongly on the selected time step; the above execution speed is for a time step of 0.002 seconds.

There are several modifications that could be made to the open-loop F/A-18 simulation to decrease its CPU and memory requirements. Most modifications to reduce the memory requirements would involve more I/O operations as the program executes. The extra programming effort and decreased CPU efficiency are deemed not worth any memory savings, especially on the MicroVAX. However, scheduling the simulation subroutine calls to the engine and steady aerodynamics models to occur at time steps which are more appropriate to their dynamics (i.e., every 0.01 seconds versus every 0.002 seconds) could result in significant CPU savings. The RFA unsteady aerodynamics coefficients are linearly interpolated as a function of Mach number. Scheduling this interpolation with the engine and steady aerodynamics could also result in additional efficiency.

Future Research

The current implementation of the equations of motion in the simulation model building framework has several known deficiencies which require attention. One of these is that "actuator reversibility" – the effect of the aerodynamic loads on control surface actuator dynamics – is not currently included in the equations. A second deficiency in the current equations of motion is that engine gyroscopic effects are not included; these effects are expected to be significant for thrust-vectored aircraft maneuvering at low-speed, high-α regions of the flight envelope.

Current tools that compute unsteady aerodynamics for aircraft simulation are generally only valid for α in the linear region of the lift-curve slope. Analyzing full-envelope fighter aircraft dynamics requires accurate prediction and exploitation of ill-behaved, complex unsteady aerodynamics which were previously dismissed as relatively unimportant.[1] Also, as computation time decreases to practical levels and more validation studies are completed, it will be increasingly important to interface recently developed nonlinear transonic unsteady aerodynamic force calculations with structural dynamics and control system equations.[4]

The effects on structural model quality that result from manipulating a "dirty" model to satisfy the practical mean-axis conditions[10] have not been quantified and require study. This manipulation of the structural model was described in the "Data Requirements and Information Flow" section.

Thrust-vectoring models made available to date do not account for the interactions between the vectored exhaust and the local flow over the tail. These interactions could be significant for aircraft like the F/A-18, where small changes in the downwash field at the horizontal stabilizer impact the entire configuration. Developing tools that accurately compute these interactions to yield useful parameters for aircraft simulation remains a challenge.

The nonlinear inertial coupling terms developed during this research[9] should be investigated for other aircraft configurations to fully evaluate their importance. If it is determined that the nonlinear coupling terms are important in elastic aircraft dynamics analysis, then the separate component effects of the $[\Delta J]_j$, $[\Delta^2 J]_{jk}$, and $\underline{h}_{jk}$ terms should be quantified.

Concluding Remarks

The authors realize that the methods and techniques presented in this paper are but one of many possible routes to build higher fidelity simulation models in a shorter amount of time. Further, there are many unanswered questions and unresolved problems in the work that has been completed thus far (see above section).

However, development of the simulation model building framework offers several new opportunities. We can now build batch simulation models in parallel with RTS models. This gives us an "independent" check simulation which can be used for: debugging outside the RTS environment; evaluating new aerodynamic, engine, or structural models before implementation in the RTS model; and post-session analysis of RTS data. Ultimately, one can imagine a scenario where control of the aircraft "math model" software resides with the test engineer's organization, while the RTS organization maintains control over the real-time interface software and the various motion-bases, cabs, and other hardware.

A second opportunity presented by development of the simulation model building framework is that control law

algorithm researchers can apply their algorithms to "real" nonlinear dynamics problems in a timely manner. This may help bridge the gap between control theoreticians and control law designers by giving the theoreticians an opportunity to sufficiently develop and demonstrate applications of their techniques which can be transferred to industry.

A third opportunity presented is the development of "control design" metrics which can be used to evaluate a control system design by properly exercising a batch simulation model. These control design metrics could include: control power requirements; agility measures (point and shoot, maneuver, transition); controller specifications; flying qualities criteria; and pilot ratings from pilot models. The "proper exercise" of a simulation could include executing "canned" maneuvers, supplying a time history of problematic pilot inputs, or using the simulation to drive a pilot model. Proper development and use of these control design metrics could powerfully augment the use of piloted RTS as an aircraft dynamics analysis tool. The metrics could be applied to the parallel batch simulation to analytically determine the regions of interest in a proposed test matrix. This would focus the piloted RTS effort on the test points which require a real pilot-in-the-loop for proper analysis.

Appendix A – Description of extended-DATAFIX File Format

The DATAFIX file format describes two sets of data arrays. The first set defines the breakpoints for each independent variable array. The extended-DATAFIX format permits more than one data array for a given independent variable. If, for example, two different angle-of-attack (α) schedules were used in measuring or computing lift coefficient (C_L) and drag coefficient (C_D), then there would be two data vectors $\{\alpha_1\}$ and $\{\alpha_2\}$.

The second set of data arrays defines the function values. The function value arrays may be up to five-dimensional, i.e. they may represent a function of five of the independent variable arrays.

The general format of the extended-DATAFIX file used as input to FITDFX is as follows:

1. COL 1-80
   File header descriptor
2. COL 1-6
   "$XDATA" keyword to signal start of independent variable arrays
3. COL 1-20
   20-character format specification for reading the array data. This format specification is referred to as FORM in subsequent discussion.
4. COL 1-6 COL 11-20 COL 31-36
   XDNAMi NXAi XVNAMi
   a) XDNAMi - name of the i^{th} independent variable data array
   b) NXAi - number of data points in array
   c) XVNAMi - generic variable name. If this field is blank then XDNAMi will be used as the default.

   This option is useful if there is more than one data array for an independent variable. Using the above description of multiple α schedules as an example,

COL 1-6	COL 11-20	COL 31-36
ALFDG1	10	ALFDG
ALFDG2	15	ALFDG

   "ALFDG1" is a 10-element data vector, and "ALFDG2" is a 15-element data vector, each containing an α schedule associated with a function or multiple functions. "ALFDG" is the variable name used in the appropriate code primitives. The engineer does not have to worry about which independent variable schedule is to be used by a given function; the FITDFX software keeps track of this information automatically.
5. NXAi data values, in ascending order, which will be read according to format specification FORM.
6. Repeat lines 4 and 5 for each independent variable; cannot exceed 100 independent variables.
7. COL 1-6
   "$FDATA" keyword to signal end of independent variable data and beginning of function value data.
8. COL 1-6 COL 11-16 COL 21-26 COL 31-36 ...
   FDNAMi XDNAMi$_1$ XDNAMi$_2$ XDNAMi$_3$
   COL 41-46 COL 51-56
   XDNAMi$_4$ XDNAMi$_5$
   a) FDNAMi - name of i^{th} function value array
   b) XDNAMi$_1$ - name of first independent variable array for FDNAMi
   c) XDNAMi$_2$, XDNAMi$_3$, XDNAMi$_4$, XDNAMi$_5$ - names of second through fifth independent variable arrays, respectively. These four names are optional and if none should be blank fields
9. Function value data for FDNAMi. The data will be read in groups of each independent variable breakpoint. The first independent variable will vary fastest. Groups will typically occupy more than one line; each group must start a new line.
10. Repeat lines 8 and 9 for each function value array; cannot exceed 300 arrays.

Appendix B – Description of JDATA File

A JDATA file is typically created by postprocessing information generated by a structural analysis tool such as EAL. The quantities in the file include:

1) Number of symmetric/antisymmetric elastic and control modes.
2) Modal frequencies in radians per second.
3) Aircraft mass from structural model, slugs.
4) Undeformed inertia matrix, $[J_o]$.
5) Generalized modal mass matrix
6) First partial derivative of the inertia matrix, $[\Delta J]_j$.
7) Second partial derivative of the inertia matrix, $[\Delta^2 J]_{jk}$.
8) Residual mass coupling between angular and elastic momentum, $\underline{h}_{jk}$.

This information is loaded into the simulation at the beginning of each run. It is likely that this file will be extended in the future as future research defines the quantities required to compute actuator reversibility effects (see "Future Research").

Appendix C – Description of SPDATA File

An SPDATA file is typically created by postprocessing information generated by the ISAC programs. The quantities in the file include:

1) Number of aerodynamic lags used in the RFA formulation which generated the data.
2) Values of the aerodynamic lags in units of reduced frequency.
3) Mach numbers at which the RFA coefficients have been computed.
4) For each Mach number, the RFA coefficients. This data is formatted in a DATAFIX-like structure since a linear table lookup is performed on the RFA coefficients with Mach number.

This information is loaded into the simulation at the beginning of each run. This file will be modified in the future to increase its utility and as future research defines the quantities required to compute actuator reversibility effects (see "Future Research").

Appendix D – Description of LOADS File

A LOADS file is typically created by postprocessing information generated by a structural analysis tool such as EAL. The quantities in the file are load coefficients and an associated integer index arranged on a line as follows:

$$\mathrm{INDX}_j \quad \mathrm{L1}_j \quad \mathrm{L2}_j \quad \mathrm{L3}_j \quad \mathrm{L4}_j \quad \mathrm{L5}_j \quad \mathrm{L6}_j$$

where

INDX_j	Integer index = 1, j^{th} mode is antisymmetric = 2, j^{th} mode is symmetric
$\mathrm{L1}_j$	In-plane/lateral shear due to mode j
$\mathrm{L2}_j$	Out-of-plane/vertical shear due to mode j
$\mathrm{L3}_j$	Axial load due to mode j
$\mathrm{L4}_j$	Out-of-plane/vertical bending due to mode j
$\mathrm{L5}_j$	In-plane/lateral bending due to mode j
$\mathrm{L6}_j$	Torsion due to mode j

Each line of data is repeated j times for each desired load point on the structure. Load coefficients for up to 10 load points can be included. Typically, three load points are included: the left wing root; the left stabilator root; and the left vertical fin root.

This information is loaded into the simulation at the beginning of each run. The load coefficients $\mathrm{L1}_j$ through $\mathrm{L6}_j$ are multiplied by their corresponding modal deflection, η_j, and are summed over the j modes to calculate the total load at a particular point.

References

1. Mabey, D. G.; and Chambers, J. R.: Unsteady Aerodynamics -- Fundamentals and Applications to Aircraft Dynamics. AGARD-AR-222, May 1985.
2. Miller, G.; Wykes, J.; and Brosnan, M.: Rigid Body-Structural Mode Coupling on a Forward Swept Wing Aircraft. AIAA Paper 82-0683, May 1982.
3. Schwanz, R. C.: Consistency in Aircraft Structural and Flight Control Analysis. AGARD-CP-228, paper #5, April 1977.
4. Hood, R. V.; Dollyhigh, S. M.; and Newsom, J. R.: Impact of Flight Systems Integration on Future Aircraft Design. AIAA Paper 84-2459, October 1984.
5. Mackall, D. A.: Qualification Needs for Advanced Integrated Aircraft. AIAA Paper 85-1865, August 1985.
6. *Advanced Continuous Simulation Language (ACSL) Reference Manual, Fourth Edition.* Mitchell and Gauthier Associates, Concord, MA, 1986.
7. Radovcich, N. A.: Preliminary Aeroelastic Design of Structures (PADS) Methods Development and Application. AGARD paper presented at 56th Structures and Materials Panel Meeting, April 1983.
8. Harschburger, H.E.; and Moomaw, R.F.: Experience with the F/A-18 Digital Flight Control System. Presented at the IEEE/AIAA 5th Digital Avionics Systems Conference, Seattle, WA, Oct. 31-Nov. 3, 1983.
9. Buttrill, C. B.; Zeiler, T. A.; and Arbuckle, P. D.: Nonlinear Simulation of a Flexible Aircraft in Maneuvering Flight. AIAA Paper 87-2501, August 1987.
10. Waszak, M. R.; and Schmidt, D. K.: On the Flight Dynamics of Aeroelastic Vehicles. AIAA Paper 86-2077, August 1986.
11. Whetstone, W. D.: *EISI-EAL Engineering Analysis Language Reference Manual.* Engineering Information Systems, Inc., San Jose, CA, July 1983.
12. Rodden, W. P.; and Love, J. R.: Equations of Motion of a Quasi-Steady Flight Vehicle Utilizing Restrained Static Aeroelastic Characteristics. *Journal of Aircraft*, Vol. 22, No. 9, September 1985, pp. 802-809.
13. Peele, E. L.; and Adams, W. M., Jr.: *A Digital Program for Calculating the Interaction Between Flexible Structures, Unsteady Aerodynamics, and Active Controls.* NASA TM-80040, 1979.
14. Adams, W. M., Jr.; Tiffany, S. H.; Newsom, J. R.; and Peele, E. L.: *STABCAR - A Program for Finding Characteristic Roots of Systems Having Transcendental Stability Matrices.* NASA TP-2165, June 1984.
15. Tiffany, S. H.; and Adams, W. M., Jr.: Nonlinear Programming Extensions to Rational Function Approximations of Unsteady Aerodynamics. AIAA Paper 87-0854, April 1987.
16. Mitchell, E. E. L.: Techniques for the Specification and Operation of a Standard Missile Simulation. Report BA-7786, Raytheon Corporation, December 1973.

Table I – Variable Nomenclature

Base	*Units*	*Description*
PHI	rad	Euler bank angle
THE	rad	Euler pitch angle
PSI	rad	Euler yaw angle
ALF	rad	angle-of-attack
BET	rad	angle-of-sideslip
GAM	rad	flight path angle
VT	ft/sec	total vehicle velocity
MACH	n.d.	Mach number
TH	lbs	total thrust
U	ft/sec	body-frame velocity, x-direction
V	ft/sec	body-frame velocity, y-direction
W	ft/sec	body-frame velocity, z-direction
P	rad/sec	body-frame velocity, about x-axis
Q	rad/sec	body-frame velocity, about y-axis
R	rad/sec	body-frame velocity, about z-axis
H	ft	altitude, above MSL
X	ft	inertial position in 'x-direction'
Y	ft	inertial position in 'y-direction'
CXX	n.d.	direction cosine
CXY	n.d.	direction cosine
CXZ	n.d.	direction cosine
CYX	n.d.	direction cosine
CYY	n.d.	direction cosine
CYZ	n.d.	direction cosine
CZX	n.d.	direction cosine
CZY	n.d.	direction cosine
CZZ	n.d.	direction cosine
E0	n.d.	quaternion
E1	n.d.	quaternion
E2	n.d.	quaternion
E3	n.d.	quaternion
RHO	slugs/ft^3	density
A	deg	aileron
K	deg	canard
E		engine
F	deg	trailing edge flap
N	deg	leading edge flap
R	deg	rudder
S	deg	stabilator (horiz tail) / elevator
SB	deg	speed brake
PLA	deg	total throttle position

Special	*Units*	*Description*
DG2RA	n.d.	deg to rad conversion factor
QBAR	lbs/ft^2	dynamic pressure
RA2DG	n.d.	rad to deg conversion factor

Prefix	*Level*	*Description*
C	1	control 'surface' command
CS	1	trigonometric cosine
D	1	control 'surface' deflection
L	2	denotes logical variable
R	1	ratio
SN	1	trigonometric sine

Suffix	*Level*	*Description*
0	2	intermediate value of state
2	2	squared
A	2	aerodynamic
AS	3	asymmetric
CG	2	vehicle center-of-gravity
D	1	d/dt, time rate of change
DG	2	variable in units of deg or deg/scc
E	2	engine, thrust related
I	2	inertial
IC	5	state variable i.c. at trim
L	4	left
LL	2	lower limit
LM	2	limit
LN	5	dummy variable for linearization
MN	2	minimum
MX	2	maximum
R	4	right
RA	2	variable in units of rad or rad/sec
RF	3	reference quantity
SE	2	sensor
SL	2	sea level
SY	3	symmetric
TR	5	non-state variable i.c. at trim
UL	2	upper limit
WG	2	wind gust
X	2	body-frame component in x-direction
Y	2	body-frame component in y-direction
Z	2	body-frame component in z-direction
ZR	2	referenced to an i.c. or trim state

FLIGHT SIMULATION OF MULTIPLE AIRCRAFT AND HELICOPTERS USING A SINGLE HIGH-PERFORMANCE COMPUTER

R. M. HOWE*

The University of Michigan, Ann Arbor, Michigan
Applied Dynamics International, Ann Arbor, Michigan

Abstract

This paper describes the simultaneous simulation of multiple airframes in real time using a single high-speed computer, the AD 100. Examples include the simulation of up to 32 aircraft over their full flight envelopes, 8 helicopters with each helicopter simulated using the blade element method, the simultaneous simulation of multiple fighter aircraft together with six-degree-of-freedom missile simulations, and the simultaneous simulation of airframes and turbofan engine dynamics. A simple method for handling separate integration mode control (reset, operate, hold) for each of the airframes is also described.

1. Introduction

Flight simulation of the engagement of multiple piloted aircraft for both training and weapons system evaluation is becoming widespread. Full flight envelope simulation of high-performance aircraft requires large numbers of multivariable aerodynamic functions, which can consume considerable processor time on general purpose computers. If the flight control system is also included in the simulation, the required integration frame rates for satisfactory dynamic accuracy can become quite high. The net result is that a single, comprehensive airframe simulation can tax the speed capabilities of very fast general purpose computers. In fact, there are a number of examples where two or more general purpose computers have been required for the real-time simulation of a single aircraft. It follows that the simulation of multiple airframes can require a large number of general purposes computers, which in turn can introduce programming and timing problems, especially in a real-time environment.

In this paper we describe the use of the Applied Dynamics AD 100 computer for the simulation of airframes. The AD 100 is a multiprocessor with special architecture and software which has been optimized for the solution of ordinary, nonlinear differential equations. The computer uses emitter-coupled (ECL) logic with floating-point word lengths of 56 bits and 65 bits. It can perform 10 million multiplies and 10 million adds per second. The total time required for a 56 bit floating point multiply is 0.075 microseconds, and for a 65 bit floating point add is 0.1 microseconds. The eqivalent overall instruction rate exceeds 100 million instructions per second. Because of the short multiply and add times, and the special architecture, the AD 100 is extremely fast in the solution of scalar problems. The Function Memory Unit (FMU) in the AD 100 includes a solid-state memory of 2.097 million 65 bit words and is designed to be especially efficient in the generation of multivariable functions. The AD 100 interface can handle up to 5 megawords per second. In the examples which follow we will see how the above characteristics can be utilized in the real-time simulation of multiple complex airframes within only one AD 100.

2. Multiple Aircraft Simulation

As a first example we consider the simulation of an aircraft with six degrees of freedom, along with a simplified flight control system. The rotational equations of motion are written using aircraft body axes, while the translational equations of motion are written using flight-path axes. [1] Quaternions are used to represent the angular orientation of the aircraft. Conventional Euler angles are computed from the quaternions for display purposes. There are 13 state variables associated with the six degrees of freedom of the rigid airframe (the 4 quaternions introduce a redundant state). The flight control system includes 7 state variables and six limiter-type nonlinearities. The count of multivariable functions used in the simulation to represent aerodynamic coefficients is the following:

2 one-variable functions
5 two-variable functions
3 three-variable functions
6 four-variable functions

The time required on the AD 100 for a single pass through the airframe equations (i.e., one integration step when using a single-pass integration method) is 128.7 microseconds. The highest frequency in the simulation is the 5 hertz frequency of the control-surface actuators. With the AB-2 (second-order Adams-Bashforth) integration algorithm, an integration step size of 8 milliseconds gives a dynamic accuracy of roughly one percent in the actuator simulation. [2] The highest frequency for the rigid airframe in our example here is the 1 hertz associated with the short-period longitudinal motion at maximim dynamic pressure. In this case the dynamic error resulting from the 8 millisecond integration step size will be much less than one percent. Since the AD 100 requires only 128.7 microseconds for one integration step, the aircraft simulation can be run at 8000/128.7 or 62.16 times

real time. The use of multiple frame-rate integration for the 5 hertz actuator loops could be used to further speed up the simulation.

It should be noted that the number of multivariable aerodynamic functions required for a full flight envelope simulation of a complex, high performance aircraft may be considerably larger than the function count listed above. Table 1 shows a tabulation of the exection times required by the AD 100 for the computation of multivariable functions using table lookup and linear interpolation. The table consists of two sections. The first section shows the execution times for the binary search required to identify the largest breakpoint contained in each function input variable. Three binary searches are mechanized simultaneously, with the total execution time for each trio dependant on the number of breakpoints. The second section lists the total execution time for function evaluation using linear interpolation. For example, the 16 aerodynamic functions listed above have 8 different input variables which require the following binary search kernals: one 65-breakpoint kernal (2.8 microseconds), one 17-breakpoint kernal (2.2 microseconds) and two 9-breakpoint kernals (3.8 microseconds) for a total execution time of 8.8 microseconds. Total execution time for the function evaluation kernals is given by $2(.6) + 5(1.1) + 3(1.9) + 6(3.5) = 33.4$ microseconds. Thus the overall AD 100 execution time for computing the 16 multivariable aerodynamic functions in our example aircraft simulation is $8.8 + 33.4 = 42.2$ microseconds. It is clear from this example and Table 1 that a much larger number of multivariable functions could be handled by the AD 100 and still have the overall integration frame time remain under several hundred microseconds.

Table 1. AD 100 execution times for multivariable function generation

Binary Search Kernals			
3 binary searches for	3 breakpoints	1.3	microsec.
3 binary searches for	5 breakpoints	1.6	microsec.
3 binary searches for	9 breakpoints	1.9	microsec.
3 binary searches for	17 breakpoints	2.2	microsec.
3 binary searches for	33 breakpoints	2.5	microsec.
3 binary searches for	65 breakpoints	2.8	microsec.
3 binary searches for	129 breakpoints	3.1	microsec.
3 binary searches for	257 breakpoints	3.4	microsec.
Function Evaluation Kernals			
Evaluation of a 1-variable function		.6	microsec.
Evaluation of a 2-variable function		1.1	microsec.
Evaluation of a 3-variable function		1.9	microsec.
Evaluation of a 4-variable function		3.5	microsec.
Evaluation of a 5-variable function		6.9	microsec.
Evaluation of a 6-variable function		13.5	microsec.
Evaluation of a 7-variable function		26.5	microsec.

It should be noted that the 128.7 microsecond frame time for the aircraft example described above includes 17.5 microseconds of overhead associated with the AD 100 simulation executive, called SIMEXEC. Thus the net execution time per integration frame is 128.7-17.5 or 111.2 microseconds. This means that each additional airframe in a multiple aircraft simulation will require only 111.2 microseconds. For illustrative purposes, let us assume that a more complex airframe simulation requires three times the 16 multi-variable functions listed at the beginning of this section (this adds 66.8 microseconds). Let us also assume that a much more complex flight control system adds another 50 microseconds, and that 100 input/output channels add still another 20 microseconds. Then the overall integration frame time for each aircraft would be $111.2 + 66.8 + 50 + 20 = 248$ microseconds. If the required real-time step size is 8 milliseconds, as assumed above, then the AD 100 could in principle simulate up to 8000/248 or 32 aircraft simultaneously in real time. As noted earlier, the use of multiple integration frame rates for the high-frequency flight control subsystems would permit an overall real-time step size considerably larger than 8 milliseconds. This would in turn increase even further the number of simultaneous aircraft which could be simulated.

3. Separate Mode Control for Individual Airframes

In a multiple engagement simulation it may be desirable to control separately in real time the integrator modes (reset, operate, hold) for each individual airframe. In the AD 100 this can be accomplished by appropriate modification of SIMEXEC. Another approach is to maintain the simulation of every aircraft in the operate mode at all times, but with the integration formulas for each state variable modified to accomplish individual mode control. To illustrate the method, assume that we are using the second-order predictor algorithm, AB-2, which is one of the most popular real-time methods. Then the difference equation for integrating the state equation, $dX/dt = F$, is given by

$$X_{n+1} = X_n + .5h*(3*F_n - F_{n-1})$$

where h is the integration step size. To allow separate mode control we modify the difference equation to the following form:

$$X_{n+1} = X_n + K1*.5h*(3*F_n - F_{n-1}) + K2*(X_0 - X_n)$$

The constants K1 and K2, in the form of real-time inputs each integration frame, control the integrator mode. Thus with $K1 = 1$ and $K2 = 0$ the integrator is in the operate mode. With $K1 = 0$ and $K2 = 0$, the integrator is in the hold mode. Finally, with $K1 = 0$ and $K2 = 1$, the integrator is in the reset mode, where in the next step the state X_{n+1} will take on the initial condition X_0. If each integration for a given airframe simulation is programmed with this formula, the integration modes for that airframe can be controlled separately from all other airframes by appropriate setting of the real-time inputs K1 and K2.

Note that in either the reset or hold mode, the state variable derivative F will be evaluated each frame using the same fixed state variables. Thus F_{n-1} will be equal to F_n. When the integration is then switched to the operate mode with $K1 = 1$ and $K2 = 0$, the difference equation for the first integration step will be given by

$$X_{n+1} = X_n + .5h*(3*F_n - F_n) = X_n + h*F_n$$

which is simply Euler integration. Subsequent steps, for which F_{n-1} in general is not equal to F_n, will revert to AB-2 integration. Actually, the usual startup method for AB-2 integration is Euler for the first step, so that the above technique should not cause any significant error in a real-time environment.

4. Multiple Helicopter Simulation

Our next example is the real-time simulation a helicopter using the blade element method. In the blade element method each rotor blade is divided into a number of segments or elements. The aerodynamic lift, drag and moment acting on each segment are calculated from the angle of attack and Mach number of the segments using two-variable aerodynamic functions. These forces and moments are then summed to obtain the overall aerodynamic forces and moments acting on the blade, which are used to integrate the blade equations of motion and to calculate the rotor forces acting on the airframe.

The angle of attack and Mach number at each blade segment are determined by computing the velocity components of the segment center of pressure with respect to the local air. These velocity components depend not only on the flapping and lagging motion of the blade, but also the rotor inflow and the translational and rotational velocities and accelerations of the airframe. The overall calculations are very computationally intensive.[3] The real-time simulation is further complicated by the need to use high integration frame rates. For reasonable performance and dynamic handling accuracy Houck has shown that 4 to 6 segments per blade are required, and that each integration step should correspond to no more than 12 to 18 degrees of azimuthal motion of the rotor.[4] For a rotor with an angular frequency of 27 radians per second this translates into an integration step size of between 8 and 12 milliseconds for the rotor simulation.

The equations of motion of the Sikorsky UH-60A helicopter have been programmed on the AD 100 computer using the Sikorsky Gen Hel engineering simulation program.[5] The simulation includes the main rotor, tail rotor, empennage, fuselage, flight controls, landing gear, engine/fuel control, and ground effects. Overall frame time on the AD 100 is approximately 1 millisecond. It follows that up to 8 such helicopter simulations could be run in real time on a single AD 100.

It should be noted here that the AD 100 uses a programming language called ADSIM, which is similar to such continuous system simulation languages as ACSL and CSSL. In the current version 5.0 of ADSIM a real-time multiple airframe simulation is limited more by the size of the ECL program memory in the AD 100 than the speed requirements. Under version 5.0 of ADSIM and with the current size of program memory in the AD 100, the UH-60A simulation described above would be limited to 4 airframes. Version 6.0 of ADSIM, which is about to be released, will support hardware subroutining as well as the software subroutining currently supported by version 5.0. This will then remove any size restrictions in the real-time simulation of multiple airframes.

5. Simulation of Multiple Airframes and Missiles

In this section we consider a comprehensive six-degree-of-freedom simulation of a tactical missile, with the possibility of simulating multiple missiles and aircraft in multiple engagements. The example missile considered here is the Hellfire, which can be launched from conventional aircraft or helicopters against surface or air targets. The system being simulated consists of the missile airframe, guidance system and flight control system, including pneumatic servos which control each of four fins. A total of 53 state variables is required in the simulation. The count of multivariable functions used in the simulation is the following:

18	1-variable functions
17	2-variable functions
7	3-variable functions

Net execution time on the AD 100 for one integration frame is 267 microseconds. The frame time required for an accurate solution in real time is four times this. Thus up to 4 Hellfire missiles can be simulated in real time on a single AD 100. More importantly, one AD 100 can be used to simulate two or three missiles as well as multiple airframes. In section 2 we saw that an integration frame time of 8 milliseconds was representative of the real-time requirement for a high performance aircraft, including its flight control and avionics system. The 1 millisecond real-time frame requirement for the missile suggests that we compute 8 integration steps for the missile per integration step for the aircraft. If we simulate 2 missiles, the required AD 100 execution time per 8 millisecond frame of the aircraft simulation is 2(277)(8) = 4432 microseconds. This leaves 8000 - 4432 = 3568 microseconds for simulating multiple aircraft. At 248 microseconds per aircraft, which is the AD 100 frame time we calculated at the end of Section 2, this would allow a simulation of up to 3568/248 or 14 aircraft in addition to the two missiles.

Consider next the simultaneous simulation of helicopters and two missiles. In Section 4 we saw that the AD 100 frame time per helicopter when using the blade element method is 1 millsecond, whereas the requirement for real time is again 8 milliseconds. Now the 3568 microseconds available per 8 millisecond frame can be used to simulate three helicopters in addition to the two missiles, all with a single AD 100.

In the above two illustrations it should be noted that the tactical missiles have a flight time of only a few seconds. When the simulated flight of a given missile has terminated, the simulation of another missile can be initiated. Thus the capability of simulating multiple aircraft as well as two missiles simultaneously in no way precludes the overall succesive simulation of many more missiles in a prolonged engagement.

6. Simulation of Airframes plus Jet Engines

Real-time high fidelity dynamic simulations of jet engines are used extensively for the design and testing of engine controllers. In high performance aircraft the engine dynamics can interact significantly with the airframe dynamics. This is especially true in helicopters, where engine interaction with the rotor often forces redesign of the engine control system. It is therefore desirable to be able to combine comprehensive simulations of airframe and engines in real time.

The AD 100 has been used extensively to simulate turbofan engines using sophisticated high-fidelity models which include compressor maps, turbine maps, etc.. Typical execution time for one integration frame for a high-bypasss turbofan is 110 microseconds. The required integration step size for a real-time simulation of the same engine is 1 millisecond. If we again use 8 milliseconds as the step size for the airframe, this suggests that we should take 8 integration steps in the engine simulation for each step in the airframe simulation. If we assume two engines per airframe, three airframes would require 2(3)(110) = 660 microseconds per integration step in simulating the engines, or 8(660) = 5280 microseconds per 8 millisecond frame for the airframes. This leaves 2720 milliseconds for simultaneous simulation of three airframes, which at 250 microseconds per airframe presents no problem for the AD 100.

7. Conclusions

We have seen how computers like the AD 100 with architecture optimized for the solution of scalar-type ordinary differential equations are capable of simulating many airframes simultaneously in real time. In particular we have shown that 30 or more six-degree-of-freedom comprehensive simulations of conventional aircraft, including flight control and avionics systems, and be simulated simultaneously in real time.

We have also shown that up to 8 helicopters can be simulated in real time using the blade element method. Engineering level six-degre-of-freedom missiles can also be simulated in real time simultaneously with the airframe simulations. Finally, comprehensive jet engine dynamic simulations can be combined with up to three real-time airframe simulations, all on a single AD 100 computer.

8. References

1. Fogarty, L.E., and R.M. Howe, "Computer Mechanization of Six-Degree-of-Freedom Flight Equations," *Simulation*, Vol. 11, No. 4, Oct., 1968.
2. Howe, R.M., "Transfer Function and Characteristic Root Errors for Fixed-Step Integration Algorithms," *Transactions of the Society for Computer Simulation*, Vol. 2, No. 4, Dec., 1968.
3. Howe, R.M., and L.E. Fogarty, "Computer Considerations for Real Time Simulation of a Generalized Rotor Model," NASA CR-2877, Sept., 1977.
4. Houck, J.A., "Computational Aspects of Real Time Simulation of a Rotary-Wing Aircraft," Masters Thesis, George Washington University, May, 1976.
5. Howlett, J.J., "UH-60A Black Hawk Engineering Simulation Program," NASA CR-166309, Dec., 1981.

Mark E. Dreier*

Bell Helicopter Textron, Inc., Fort Worth, Texas

ABSTRACT

Stringent handling qualities requirements and the recognized importance of pilot-in-the-loop evaluation of highly maneuverable helicopters requires sophisticated and accurate representation of the main rotor if simulation is to be used as an engineering design tool. This paper addresses the development of a fully coupled, real-time, blade element aeroelastic rotor model and its incorporation into a manned simulation. Design goals include real-time operation within a total piloted simulation environment, tabular stall/compressible flow blade section aerodynamics, two fully-coupled flap/lag/torsion blade modes, non-linear distributed mass, chord and blade shape, preservation of rotor harmonics and transient response, single blade failure capability, and rapid data changes. All of these goals are met through judicious use of state-of-the-art simulation computers, formality of the mathematical model, parallel computation of aerodynamic and inertial models distributed between two processors and reduction of computer frametime using pre-processed data. The rotor model, combined with an airframe, power plant and control system model constitutes a comprehensive rotorcraft simulation.

NOMENCLATURE

BMC	Bending moment coefficients
D	Deformation matrix
dF	Distributed force vector
dM	Distributed moment vector
F	Radially integrated force
f_0	Constant in forcing function
J	Inverse of trim Jacobian
M	Hub moment, Mach number
m	Distributed mass of blade
Mg	Generalized inertia
Nb	Number of blades
q	Blade position vector
q_0	$\partial(q_0)/\partial(\Theta_p)$
q1,q2	Yawed flow coefficients
qt	Blade section dynamic pressure
$T_x(\Phi)$	X axis transformation, Y and Z similar
T	Final blade axes transformation
T_0	Prescribed axes transformation
V	General velocity vector
Vc	Speed of sound
α	Angle of attack
β	Modal vector
Δ	Six DOF modeshape
Λ	Yawed flow angle
Φ	Blade angular position vector
$d\Phi_0$	$\partial(\Phi_0)/\partial(\Theta_p)$
Θ_p	Prescribed blade pitch angle
Θ_0	Reference blade pitch angle
Ψ_k	Azimuth angle of kth blade
ω	Angular velocity vector
ωn	Modal natural frequency
Ωz	Rotor spin rate vector

Superscripts

.	Differentiation in time
T	Transpose of matrix or vector

Subscripts

0	Fixed, manufactured
a	Aerodynamic entity
b	Blade axes system
c	Chordwise
h	Hub axes system
i	Inertial entity
k	Blade number
m	Mast axes system
n	Normal to blade axes
p	Prescribed by pilot input
r	radial station, radial component, rotating system
x,y,z	Direction of vector component
3	Translational components of entity
6	Rotational components of entity

INTRODUCTION

The role of man-in-the-loop simulation in the preliminary design phase of aircraft has become increasingly significant. The simulator as an engineering tool provides insight into the effects of non-linear phenomena on loads and handling qualities, and reveals interactions between subsystems and major systems when used in a

---

* Senior Computing Project Engineer Member AIAA

comprehensive rotorcraft simulation. The benefits of piloted simulation become tangible when investigation suggests modification in the design phase rather than after the prototype aircraft has been manufactured.

The various elements of the mathematical model describing the loads and motion of dynamic variables must have a high degree of fidelity and sophistication for the simulation to be useful. Rotorcraft, which fly in a wide range of conditions from hover to high forward and vertical speed, require particularly complex models to maintain fidelity. In particular, the highly non-linear aerodynamic environment in which a rotor operates and the non-uniform distribution of rotor properties presents a special problem in modelling; fidelity requires retention of detail, but to ensure real-time operation some sophistication has historically been sacrificed.

State of the Art

The compromises to rotor model fidelity usually encompass constraints on both inertial and aerodynamic calculations. The most common approaches assume the blade to be a hinged beam of uniform mass, chord and twist distribution and employ small angle assumptions in the angle of attack expressions. The aerodynamic coefficients are assumed either linear or simple curve fits of total rotor performance data, spreading the effects of stall, compressibility and reverse flow over the rotor disk. Furthermore, the flapping dynamics are often modelled using a quasi-static approach; the solution to the forced response problem being represented in the non-rotating frame with a tip path plane model. These assumptions lie at the heart of classical rotor analyses [1,2] and are used with some success in a current simulation program. [3] This approach, while good for initial performance estimation fails to model adequately the effects of aeroelastic rotor blade shapes, higher frequency modes, stall, reverse flow, compressibility or transient response. This approach also requires the modeller to expand the equations by hand so that many non-linear and higher order effects can be discarded, thus making the radial integrations tractable and expressible in closed form.

Another method that has met with some success uses specially constructed hybrid computers with analogue circuitry to model the aerodynamic and inertial portions of the rotor equations. [4] This method, which is based on a more rigorous math model, [5] permits individual blade dynamics to be modelled. The radial integrations are performed in real-time and do account for stall, compressibility, reverse flow and higher frequency modes. However, this method also has some important limitations. The blade section is assumed constant over the entire span of a rotor blade, the analogue circuitry requires a constant recalibration due to the drift inherent in analogue circuits and the cost for such a specialized computer is high.

Other methods have incorporated high speed fixed point arithmetic computers which introduce scaling problems or somewhat slower machines which necessitate reduction of the number of radial stations, aerodynamic tables, azimuth steps and/or blades to achieve real-time operation. These methods, which may be calibrated with measured performance data, severely limit their usefulness as engineering design tools.

This paper addresses the development of a real-time blade element aeroelastic rotor model which circumvents the problems described above. This model uses a formalized mathematical representation and employs two state-of-the-art simulation computers to provide a real-time simulation tool. Design goals included tabular blade section lift, drag and moment coefficients as functions of angle of attack, Mach number and radial station, two fully coupled aeroelastic blade modes, non-linear distributed blade properties, eleven radial stations, four blades, azimuth steps less than 15 degrees and real-time operation with frametimes less than 16. milliseconds.

FUNDAMENTALS OF THE ROTOR MODEL

Philosophy

The rotor model is divided into three problems - the aerodynamic loads calculations, the inertial loads calculations and the modal model. In each problem, rigorous declarations of axis systems and vectors representing blade motions and loads are made. From these definitions, fundamentals of vector mechanics are employed at the vector level to generate displacements, velocities and accelerations of blade sections whence the aerodynamic and inertial loads are calculated. The expressions are not expanded; they are left in vector form. This has three distinct advantages. It reduces the possibility of

coding and expansion errors, it makes it possible to include all terms without discrimination and it can take advantage of efficient vector and matrix level modules available in some simulation languages.

Axis System Definitions

The various axis systems, load vectors and motion vectors are specified first. All distributed property vectors are assumed to vary radially, all axis system transformations assume large angles and non-linear products are assumed to be important. The various axis systems are discussed in detail in Appendix A and are shown in figure 1.

Load Vectors

The distributed loads on a blade are described with two column vectors, dF and dM. The dF vector is composed of distributed forces at each radial station; the dM vector contains the distributed moments. They are ordered as shown:

$$dF_k = \{dFx,\ dFy,\ dFz\}^T,$$

$$dM_k = \{dMx,\ dMy,\ dMz\}^T \qquad (1)$$

where k is the blade number and the superscript T means transpose.

The dF and dM vectors are resolved to the rotating hub system. Both of these vectors are the summation of aerodynamic loads and inertial loads, dFi + dFa and dMi + dMa. These loads are radially integrated from blade root to blade tip to produce Fr and Mr which are shaft loads in the rotating system:

$$Fr_k = \int dF_k\, dr,\quad Mr_k = \int dM_k\, dr \qquad (2)$$

The Fr and Mr loads are then transformed to the mast axes and summed to produce shaft loads:

$$Fm_k = \sum_{k=1}^{Nb} Tz_k^T * Fr_k \qquad (3)$$

$$Mm_k = \sum_{k=1}^{Nb} Tz_k^T * Mr_k \qquad (4)$$

where Tz_k^T represents the z axis transformation from hub to mast axes through the combined rotor and blade spacing azimuth angle, Ψ_k. No assumption about angular spacing between blades is made. Nb is the number of blades in the rotor.

Motion Vectors

Calculation of the distributed loads requires analysis of the motion of the blades in space. The motion vectors contain displacements, rotations and the first and second time derivatives of each with respect to inertial space but resolved to hub and blade axes respectively. Specifically, the translational displacement of any station is the sum of the initial or prescribed blade shape and the elastic deflection:

$$q(r,t) = \{\ x_b,\ y_b,\ z_b\ \}^T$$

$$= q_p(r,t) + \Delta_3(r) * \beta(t) \qquad (5)$$

$$q_p(r,t) = q_0(r) + Q(r) * (\Theta_p - \Theta_0) \qquad (6)$$

The corresponding angular displacement is defined:

$$\Phi_p(r,t) = \{\ \Phi_1,\ \Phi_2,\ \Phi_3\ \}^T$$

$$= \Phi_0(r) + d\Phi_0(r) * (\Theta_p - \Theta_0) \qquad (7)$$

These prescribed angles rotate a vector from hub axes to final blade axes via the three axis transformation:

$$V_f = D(\Delta_6,\ \beta) * T_0(\Phi_1,\Phi_2,\Phi_3) * V_h \qquad (8)$$

The general three axis Euler angle transformation is detailed in Appendix B. The rotation using prescribed angles first, followed by deformation (perturbation) angles is a direct consequence of the requirement of many structural analysis codes to maintain Maxwell's reciprocity theorem in the mass and stiffness matrices. This method is used in the rotorcraft analysis program MOSTAB. [5] The Δ_3 matrix contains the translational components of all modeshapes used in the analysis. The Δ_6 matrix contains the rotational components of the same modeshapes. The β column contains all the modal participation factors for a given blade. The vectors q_0 and Φ_0 represent the initial translational and rotational displacements such as precone, sweep and twist schedule. The matrices Q and $d\Phi_0$ are the partial derivatives of the translational and rotational displacements with blade pitch, evaluated at $\Theta_p = \Theta_0$ with all cyclic pitch inputs set to zero. The terms Θ_p and Θ_0 are the actual blade pitch and reference blade pitch respectively. For the sake of convenience, the notation indicating functions of radius and/or time is dropped, the units being inferred from context.

At a given azimuth and radial station, the contributions to the local aerodynamic velocity are the fuselage motion, rotor rotational rates, aeroelastic effects and induced velocity. All these velocities are grouped in three separate velocity vectors - inertial, induced and aerodynamic. All three vectors have six degrees of freedom. The inertial velocities are due to the motion of the blade with respect to inertial space, the induced velocities are generated by changing the momentum of the air in the vicinity of a body, and the aerodynamic velocity is the vectorial sum of the other two. All three vectors are resolved to blade axes. The equations describing the aerodynamic velocity in blade axes are:

$$Va_k = T_k^*(Vha_k + \dot{q}_k + \omega ba_k \times q_k) \quad (9)$$

where Vha and ωba are the translational and rotational aerodynamic velocities in rotating coordinates and are composed of inertial and induced velocities. The rotor rotational rate, Ωz is also included in ωba. The derivative term represents the blade motion due to pilot input and elastic deformation and is given by:

$$\dot{q}_k = Q * \dot{\Theta}_{pk} + \Delta_3 * \dot{\beta}_k \quad (10)$$

The induced velocity has a pronounced effect on the loads and motion of a rotor blade. The description of the model is deferred to a later section.

Blade station acceleration is required to produce the inertial loads. These accelerations are a combination of hub accelerations, modal motion and Coriolis accelerations. For translational loads, the accelerations must be resolved to hub axes. The required expression without development is given as:

$$dVi/dt_k = \dot{Vh}_k + (\dot{\omega h}_k \times Vh_k) + \ddot{q}_k + 2(\omega b_k \times \dot{q}_k) + \dot{\omega b}_k \times q_k + \omega b_k \times (\omega b_k \times q_k) \quad (11)$$

where Vh, $\dot{V}$h and $\dot{\omega}$h are the inertial velocity, inertial acceleration and hub spin rate resolved to the rotating system. The remaining terms are defined below:

$$\omega b_k = \omega h_k + \Omega z \quad (12)$$

$$\dot{\omega b}_k = Tz(\Psi_k) * (\dot{\omega m} + \dot{\Omega z} + \omega m \times \Omega z) \quad (13)$$

$$\ddot{q}_k = Q * \ddot{\Theta}_{pk} + \Delta_3 * \ddot{\beta}_k \quad (14)$$

From these expressions, the aerodynamic and inertial loads are determined.

Aerodynamic Loads

For the kth blade, the column vector Va stores the radial, chordwise and normal velocity components resolved to the deformed blade axes at each radial station. Reference 6 describes a two dimensional strip theory for blade section aerodynamics, modified for yawed flow. Figure 2 illustrates the velocity vectors.

$$Va = \{ Ur, Uc, Un \}^T \quad (15)$$

$$\Lambda = \arctan(Ur/Uc) \quad (16)$$

$$\alpha = \arctan(Un/Uc) * \cos(\Lambda) \quad (17)$$

$$M = \sqrt{(Un*Un+Uc*Uc)}/Vc*\cos(q1*\Lambda)^{q2} \quad (18)$$

$$q_t = \rho*(Un*Un+Uc*Uc)/2 \quad (19)$$

The terms q1 and q2 are user defined data. From the angle of attack, yawed flow angle, Mach number and radial station location, the blade section lift, drag and moment coefficients can be determined using a bivariant table interpolation. The coefficients can then be modified for yawed flow if desired. With the coefficients known, the normal force, chordwise force and torsional moment can be found from:

$$dFn = -q_t*c*(\cos(\alpha)*CL+\sin(\alpha)*CD) \quad (20)$$

$$dFc = -q_t*c*(-\sin(\alpha)*CL+\cos(\alpha)*CD) \quad (21)$$

$$dMxa = -q_t*c*c*CM \quad (22)$$

Final resolution of the loads back to the hub system is achieved via the transpose of the T_k matrix:

$$dFa_k = T_k^T * \{ 0, dFc, dFn \}_k^T \quad (23)$$

$$dMa_k = T_k^T * \{dMxa, 0, 0 \}_k^T + q_k \times dFa_k \quad (24)$$

Inertial Loads

Using d'Alembert's paradox, the

inertial forces are defined by:

$$dFi_k = -m * dVi/dt_k \quad (25)$$

Inertial moments could be calculated in a fashion similar to the aerodynamic moments. However, with some manipulation, the inertial moments are formed into linear operators of the modal participation factor β:

$$Mri_k = BMC * \beta_k \quad (26)$$

Note that Mri_k is already a radially integrated result. The bending moment coefficients in the array BMC are functions of modeshapes, frequencies and mass distribution and are generated off-line.

Hub and Mast Loads

The hub forces are given by:

$$Fr_k = \int(dFi_k + dFa_k)\, dr \quad (27)$$

The aerodynamic hub moment is given by:

$$Mra_k = \int(q_k \times dFa_k)\, dr \quad (28)$$

The hub forces and moments are resolved to the non-rotating mast system as described in expressions 3 and 4. The hub roll and pitch moments use the Mri_k values. The aerodynamic moments are calculated for the torque requirements and to measure moments which would produce swirl (rotary) induced velocities.

Modal Model

The modal model is a second order differential equation with a non-linear forcing function. The forcing function is created from the same distributed load vectors that generate the hub loads. For each blade, one or more modes may be active. The modeshapes and frequencies are generated off-line by a Myklestad[7] analysis, and include the centrifugal force field. Each mode uses exactly the same modal model in form. Only the selection of the modeshape data and frequencies distinguishes the type of mode. The modal model is given below:

$$\ddot{\beta} + \omega n^2 * \beta = f(\Delta_3, dFa, dFi, f_0, \dot{\beta}, \beta, Mg) \quad (29)$$

where

$$Mg = \int (\Delta_3^T * m * \Delta_3)\, dr \quad (30)$$

The modal model is a perturbation model constructed about some initial pre-twisted, pre-coned, pre-swept blade shape, therefore the forcing function for the modal model must include only perturbation forces and must not include linear influences of β or its time derivatives. The appearance of β and its second derivative in the forcing function implies that only the non-linear influences are present. The linear terms involving the modal participation factor are removed algebraically using vector arithmetic. The term f_0 is used to account for blade loads due to the initial shape and is a constant which is calculated off-line.

Induced Velocity Model

The induced velocity model is calculated using a modified form of the Glauert momentum model.[6] The modifications allow azimuthal and radial variation of inflow as a function of advance ratio and are well defined at hover, high advance ratio and high inflow. Induced velocity and loads are coupled implicitly and in reference 3 are iterated together to achieve the balance between thrust and rate of change of momentum. In this model, such an iteration would be prohibitively costly in frametime. Therefore, the assumption is made that shaft loads remain constant during a frame and the induced velocity alone is iterated. The induced velocities are stored in the vectors Vind and ωind. The representation is sufficiently general to allow other induced velocity models, including dynamic inflow models to be incorporated easily.

IMPLEMENTATION

Rotor Model

The rotor model was programmed in the ADSIM and MPS languages which are used on the AD100 and AD10 computers respectively. These computers, produced by Applied Dynamics International, are high-speed pipelined processors which can be programmed to execute in parallel. The AD100 is a floating point processor with precision sufficient for the inertial and aerodynamic loads calculations. The AD10 is a scaled integer arithmetic processor which is very efficient in table look-up. The languages are designed to take advantage of the pipelines and the modularity of the mathematical model works particularly well with the modular program modules available in MPS and ADSIM. The computational task is thus distributed to take advantage of the

execution speeds and specialities of the processors. The process described next is best visualized by referring to figure 3.

The CONTROLS, VELOCITY and BLADE MOTION module calculates the control inputs to the rotor at the swashplate, the aerodynamic velocity, the inertial velocity and acceleration at the top of the mast and the blade shape and its derivatives resolved to the rotating co-ordinate system. The DF1 block calculates the inertial forces, the angle of attack and the mach number for a given radial station. The DF2 block finishes the distributed blade load calculations and generates the distributed forcing functions. The RADIAL, INDUCED VELOCITY and INTEGRATE MODES module produces the shaft loads, forcing functions and induced velocity and performs the modal integration in time using a state transition matrix method. As discussed above, the assumption is made that the shaft loads in one frame remain constant while the induced velocity calculation is iterated. This break in the implicit nature of the expressions has no measureable effect on the dynamic loads or induced velocity provided the iteration on induced velocity is performed. Since the induced velocity model is a small fraction of the total computational task, it does not contribute significantly to the frame-time but does provide the necessary balance between thrust and induced velocity. At the bottom of the dynamic loop, the rotor loads and induced velocities are transferred to the airframe model for additional processing.

The parallelism and distribution of computational tasks between the AD100 and the AD10 is not required, but certainly helps to achieve the real-time capability. While the AD10 is performing the aerodynamic coefficient look-up for all radial stations for the kth blade, the AD100 is performing the initial angle-of-attack and mach number calculations for the k+1 blade and finishing the loads calculations for the k-1 blade when the pipeline is fully loaded. Again, vectorization makes this task easy to mechanize.

Inner loop, outer loop and transfer variables were chosen to minimize the amount of calculation and inter-computer I/O. Whenever possible, all calculations using input data which generate constants were pre-processed.

Airframe, Engine and Control System Models

The rotor is interfaced to airframe, engine and control system models. The fuselage is represented as a six degree of freedom rigid body [3] with a flexible mast and pylon. [8] The airframe aerodynamic model is a non-linear, table assisted model of a fuselage, wings and stabilizer. The tail rotor is modelled in closed form analytical expressions for thrust, H and Y forces, tail rotor hub moments and torque. The powerplant model includes two fully independent engines and fuel controls which feed the drive train model. This permits one and two-engine inoperative capability. Both the powerplant and airframe models are programmed in ADSIM and MPS and are executed in the same frame as the rotor. The control system model is programmed in FORTRAN and executes in parallel with the AD100 and AD10 on a VAX 8600 which serves as the simulation executive computer.

Trim Procedure

A "fly-to-trim" procedure which uses the speed of the AD100 and the reliability of the Newton-Raphson method is employed. In seven non-real-time passes through the dynamic loop, a Jacobian matrix relating average accelerations of the rigid body over one rotor period to the swashplate angles and fuselage Euler angles Φ_e and Θ_e (or Θ_e and Ψ_e) is calculated. In real-time mode and in the spirit of a Newton-Raphson iteration scheme, the unbalanced accelerations at any moment are premultiplied by the inverse of this Jacobian (J) and integrated. This result is fed back to the control system. Figure 4 demonstrates the concept.

The result is a so-called wash-out function. Though the acceleration vector is now time-varying and will produce a time variant feedback signal, the transient terms are well attenuated even for a two bladed rotor and the aircraft will "fly" to trim after only a few seconds. In the current model, this method requires only 4 to 8 seconds after the calculation and inversion of the Jacobian to trim the fuselage and blade state variables.

Additional Interfaces

A Simulation Executive System executing on the simulation executive computer directs the activities of the rotorcraft simulation. The Simulation Executive System is used to provide communications to and from the AD10, AD100, ADI host (a VAX 11/780), the visual system, cockpit

graphics system, the sound generator and the signal conversion unit. It also executes a portion of the mathematical model. The basic configuration is shown in figure 5.

VALIDATION, CORRELATION AND RESULTS

Correlation and validation of the rotor model was performed in two distinct phases. The inertial model was validated using rigorous and well known results from the physics of a whirling beam. Closed form solutions were compared against the time integrated results of the rotor code. The aerodynamic model was validated using linear lift, drag and moment coefficient tables. The rotor code results were compared against classical closed form solutions, then against the more rigorous model of reference 6, namely C81. Control sweeps of collective and cyclic input, and airspeed sweeps from hover to high forward and high vertical speed were conducted. Detailed examination of the distributed aerodynamic parameters as well as gross rotor performance and modal response were made. In all regimes tested, the loads and motion of this rotor demonstrate very good agreement with C81. Figures 6 through 17 demonstrate typical correlation results at 120 knots. These results were achieved without the use of empirical corrections beyond what the induced velocity model introduced.

The primary objective of this effort was real-time operation in a piloted, full rotorcraft simulation. Table 1 shows the current frametime and azimuth increments for this simulation.

Table 1 Simulation module frametimes

Module	Frametime (msec.)	Azimuth Step (°)	Calculations Per Frame
Rotor	3.42	7.14	1
Fuselage	1.80	----	1
Combined	5.00	10.44	1
Total Simulation	16.0	33.41	1
		16.70	2
		11.14	3

It is interesting to note that when the standalone rotor frametime and fuselage frametime are summed, the result is greater than the combined model. This is a fortuitous consequence of the ADSIM compiler which performed additional optimization on the combined model. Table 1 also shows the performance of the simulation using a 16.0 millisecond frame, ideal for the state-of-the-art visual systems. Three azimuth step values are given, based on the number of model frames calculated per simulation frame. As can be seen, 2 model frames per simulation frame almost meets the design goal of 15 degree azimuth steps; 3 model frames per simulation frame easily achieves the goal. The times shown in table 1 are for a rotor which is operating with two fully coupled flap/lag/ torsion modes for each of four blades. Each radial station on each blade uses its own airfoil tables with 65 angle of attack breakpoints and 14 mach number breakpoints for the CL, CD and CM functions. The fuselage and engine models also employ similar tables.

CONCLUSIONS

A sophisticated blade element aeroelastic rotor model which operates in real-time has been developed. The vector methodology, modular approach and formal representation in its design permit easy integration with a non-linear airframe model. The entire aircraft model is incorporated in a comprehensive rotorcraft simulation which uses the speed of specialized simulation computers and distributed processing to maximum advantage. Correlation with an industry standard simulation code shows excellent agreement in a wide range of flight regimes.

ACKNOWLEDGEMENTS

The author acknowledges the support of Messers Alan Myers, Robert Fortenbaugh, Richard Bennett, Richard Oris, Marlon Gordon and Matt Landry during this project.

APPENDIX A

Axis Systems Definitions

All axis systems used in this model are right-handed Cartesian. The fundamental axis system is the inertial axis system. It is fixed in space and provides the base from which to measure absolute values of displacement, velocity and acceleration. The second system, attached to a reference point on and moving with the aircraft, is known as the fuselage system. It is customary, but not mandatory for the origin of this system to be attached to the aircraft center-of-gravity. The X_f axis lies parallel with the waterline and is pointed forward, the Y_f axis points toward the right wing and the Z_f axis is directed toward the floor. This system moves with

six degrees of freedom relative to inertial space; three angular and three translational. The angular displacements are measured with the Euler angles Φ_e, Θ_e and Ψ_e. The third system is the mast axis system which moves with six degrees of freedom relative to the fuselage system but does not spin with the rotor. The X_m and Y_m axes lie in a shaft normal plane with origin at the top of the shaft. The X_m axis generally points forward, the Z_m axis is parallel and coincident with the shaft. The fourth system is the hub axis system which shares a common origin with and is initially coincident with the mast system. The hub system spins with the rotor. The fifth system is a reference line marking selected radial stations on the blade. In this model, the blade section center-of-gravity and aerodynamic center are assumed coincident; the reference line is used to locate the aerodynamic centers. This reference line is resolved to the hub system. The last system is a collection of radially distributed blade section systems. Each blade section system can move with six degrees of freedom relative to the hub axes. The axes are aligned such that the X_b axis lies tangent to the reference line and points generally toward the hub, the Y_b axis lies parallel with the chord line and the Z_b axis completes the triad. Translational displacements are measured with respect to the hub axes. Angular displacements are measured with Euler angles Φ_1, Φ_2 and Φ_3 about the X_b, Y_b and Z_b axes. Figure 1 in the main text shows the relationship of these various axis systems.

APPENDIX B

General Euler Angle Transformations

The convention for Euler angle transformations will rotate a given axis system first through an angle Φ_3 about the Z axis, then about the new intermediate Y axis through the angle Φ_2 and finally about the newer intermediate X axis through the angle Φ_1. All three rotations can be conveniently expressed in matrix notation with the single matrix T_0, defined as:

$$T_0(\Phi_1,\Phi_2,\Phi_3) = T_x(\Phi_1) * T_y(\Phi_2) * T_z(\Phi_3) \tag{B1}$$

$$T_z(\Phi_3) = \begin{vmatrix} \cos(\Phi_3) & \sin(\Phi_3) & 0 \\ -\sin(\Phi_3) & \cos(\Phi_3) & 0 \\ 0 & 0 & 1 \end{vmatrix} \tag{B2}$$

$$T_y(\Phi_2) = \begin{vmatrix} \cos(\Phi_2) & 0 & -\sin(\Phi_2) \\ 0 & 1 & 0 \\ \sin(\Phi_2) & 0 & \cos(\Phi_2) \end{vmatrix} \tag{B3}$$

$$T_x(\Phi_1) = \begin{vmatrix} 1 & 0 & 0 \\ 0 & \cos(\Phi_1) & \sin(\Phi_1) \\ 0 & -\sin(\Phi_1) & \cos(\Phi_1) \end{vmatrix} \tag{B4}$$

Thus, resolving a vector in hub axes to blade axes is represented simply as:

$$V_b = T_0(\Phi_1,\Phi_2,\Phi_3) * V_h. \tag{B5}$$

If elastic deformations are assumed small compared to initial angular deformations such as precone, twist and sweep, then the V_b vector can be resolved to final (deformed) blade axes through the deformation matrix D, defined as:

$$D(r,t) = \begin{vmatrix} 1 & \delta_z & -\delta_y \\ -\delta_z & 1 & \delta_x \\ \delta_y & -\delta_x & 1 \end{vmatrix} \tag{B6}$$

where $\{ \delta_x, \delta_y, \delta_z \}^T = \Delta_6(r) * \beta(t)$ (B7)

and Δ_6 represents the angular portions of the modeshapes and β is the modal participation factor. Thus

$$V_f = D * T_0 * V_h = T * V_h \tag{B8}$$

Note that perturbation angles are not added directly to the prescribed shape to effect the final transformation. They instead perform another series of rotations to achieve the final orientation.

REFERENCES

1) Gessow, A. and Myers, G. C., "Aerodynamics of the Helicopter", 1st ed., Frederick Ungar Publishing Co., 1952

2) McCormick, B. W., Jr., "Aerodynamics of V/STOL Flight", Academic Press, New York, 1967

3) Talbot, P. D., Tinling, B. E., Decker, W. A. and Chen, T. N., NASA TM 84281, 1982

4) Hoffman, J. A. and Thoren, R. J., "Mathematical Models and Hybrid Program for the Special Purpose Rotorcraft Simulator (SPURS) Baseline System", Vol 1., Paragon Pacific, Inc., PPI-5505-2, May, 1978

5) Hoffman, J. A., "Analysis Methods Incorporated in the MOSTAB-HFA Computer Code", Paragon Pacific, Inc., PPI-1015-2, Sept., 1975

6) McLarty, T. T., "Rotorcraft Flight Simulation with Coupled Rotor Aeroelastic Stability Analysis", Vol 1., USAAMRDL-TR-76-41A, 1977

7) Sadler, S. G. and Ellis, D. B., "Documentation of Myklestad Analysis (DNAM06)", Bell Helicopter Textron, Inc. 299-099-608, 1977

8) Fortenbaugh, R. L. and Rossi, J. M., "Incorporation of a Flexible Support System into the ARMCOP Mathematical Model for Real-Time Piloted Helicopter Simulation", Bell Helicopter Textron, Inc., 1119, Jan. 1987

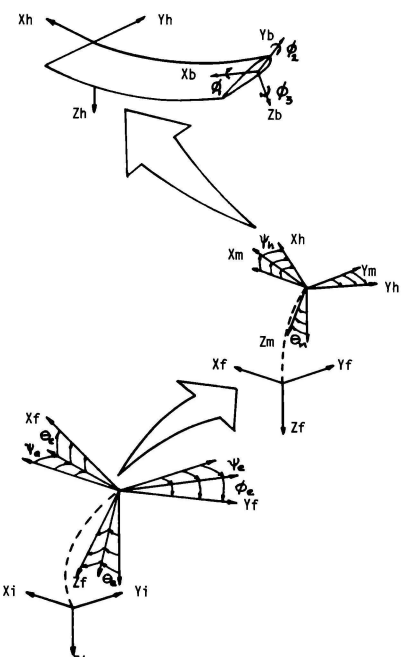


Figure 1 Axis systems

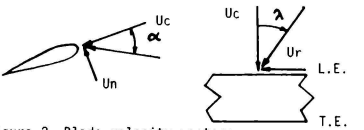


Figure 2 Blade velocity vectors

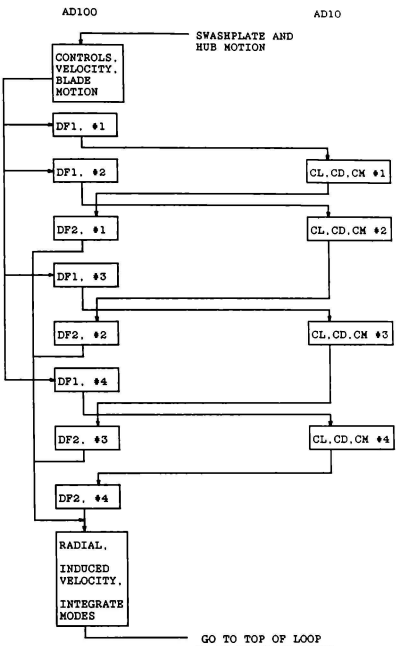


Figure 3 Rotor model flowchart

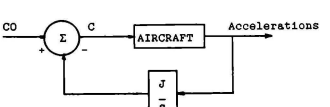


Figure 4 Fly-to-trim flowchart

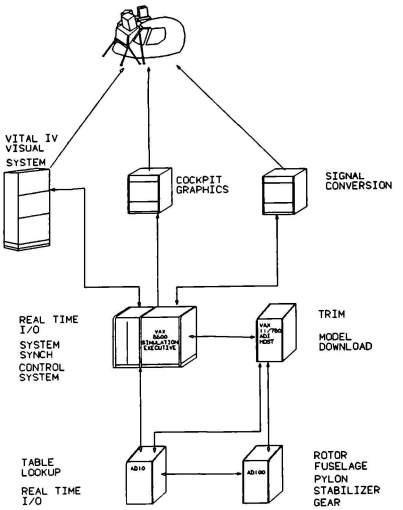


Figure 5 Overall simulation configuration

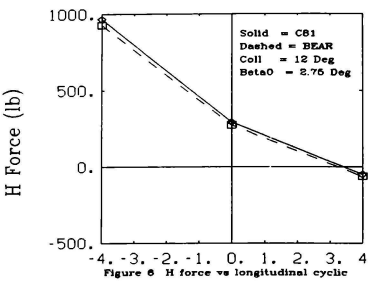


Figure 6 H force vs longitudinal cyclic

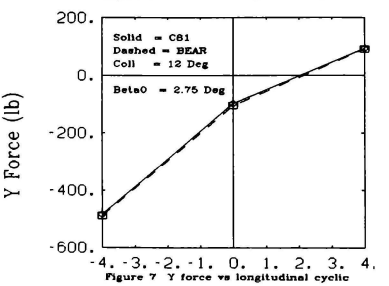


Figure 7 Y force vs longitudinal cyclic

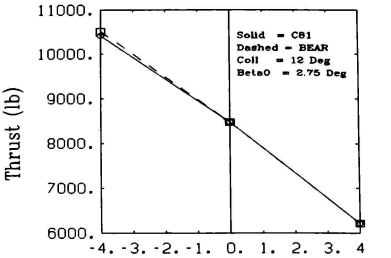


Figure 8 Thrust vs longitudinal cyclic

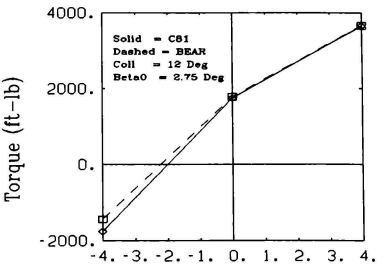


Figure 9 Torque vs longitudinal cyclic

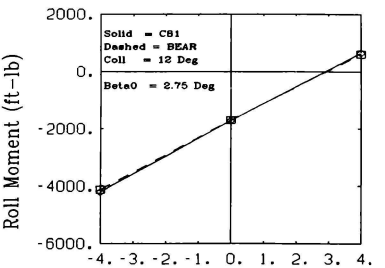


Figure 10 Roll moment vs longitudinal cyclic

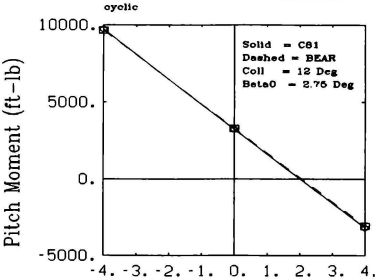


Figure 11 Pitch moment vs longitudinal cyclic

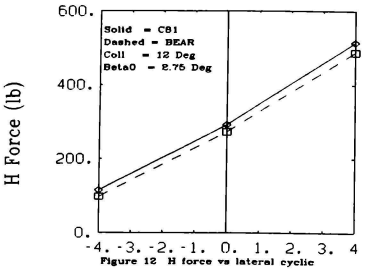


Figure 12 H force vs lateral cyclic

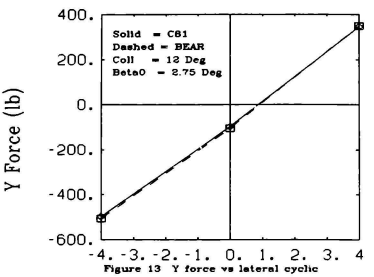


Figure 13 Y force vs lateral cyclic

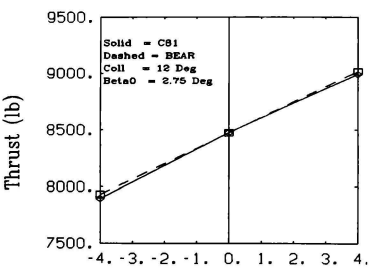


Figure 14 Thrust vs lateral cyclic

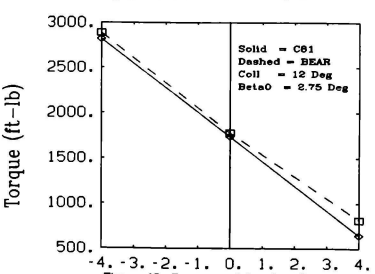


Figure 15 Torque vs lateral cyclic

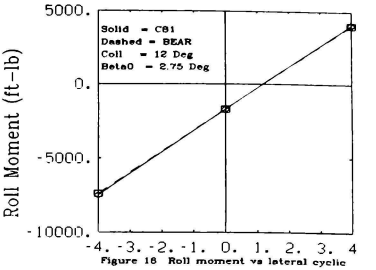


Figure 16 Roll moment vs lateral cyclic

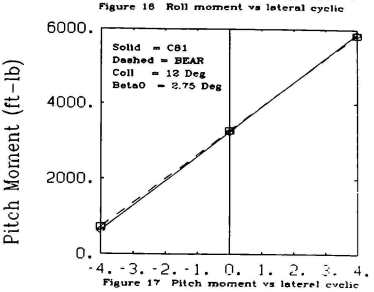


Figure 17 Pitch moment vs lateral cyclic

NONLINEAR SIMULATION OF A FLEXIBLE AIRCRAFT IN MANEUVERING FLIGHT

Carey S. Buttrill*
NASA Langley Research Center

Thomas A. Zeiler**
PRC Kentron Inc.

P. Douglas Arbuckle*
NASA Langley Research Center

Abstract

A math model integrating nonlinear rigid-body flight mechanics and linear aeroelastic dynamics is described. The equations of motion for an elastic aircraft in accelerated flight are developed using Lagrangian mechanics. Modes of undamped free vibration that satisfy the first-order mean-axes constraints are used as generalized coordinates. Terms in a modal formulation that define nonlinear inertial coupling between total angular momentum and elastic momentum are identified.

A simulation model of an F/A-18 (configured with tip missiles) that includes angular/elastic inertial coupling has been constructed. The effect of this inertial coupling on open-loop rigid-body simulation response was found to be negligible. The effect on the elastic response of the model is generally small. Exceptions occurred, but the elastic modes significantly affected by inertial coupling were aerodynamically decoupled from the rest of the model. This aerodynamic decoupling is typical of, but not guaranteed for, aircraft.

The elastic modes affected by inertial coupling are those modes which induce changes in total aircraft mass distribution. The elastic effect is noticable if deformation-induced mass distribution changes are significant with respect to modal mass and modal frequencies. A modal parameter is presented that characterizes the level of inertial coupling between elastic momentum and rigid-body angular momentum.

Nomenclature

Symbols

$\underline{d}_i$	translational deformation at node i, ft
dm_i	lumped mass at node i, slugs
$[dI]_i$	lumped inertia at node i, slug-ft^2
$\underline{F}$	vector of total applied force on the aircraft, lbs
$\underline{g}$	gravitational acceleration vector, ft/sec^2
$\underline{h}_{jk}$	3×1 vector representing energy coupling between rigid-body angular velocity and modal velocity in mode k due to deformation in mode j, slug-ft^2, equation (14)
$[J]$	inertia matrix of the total aircraft in the deformed condition, slug-ft^2, equation (17)
$[J_o]$	inertia matrix of the total aircraft in undeformed reference condition, slugs-ft^2, equation (10)
$\underline{L}$	vector of total applied moment on the aircraft, ft-lbs
m	total mass of the aircraft, slugs
M_{jk}	generalized mass coupling between elastic modes j and k, slug-ft^2
p	roll rate, rad/sec
$\underline{P}$	position vector locating the body-frame origin in the local Earth frame, ft
q	pitch rate, rad/sec
Q_{η_j}	generalized force on elastic mode j, ft-lbs
r	yaw rate, rad/sec
$\underline{r}_i$	position vector locating the center of the lumped mass i with respect to the body-frame origin when the body is in the undeformed reference condition, ft
$\underline{R}_i$	position vector locating the center of lumped mass i in the Earth frame, ft
$R_j(\underline{\omega})$	parameter characterizing the quasi-steady response of mode j to rigid-body rotation rates, nondimensional, equation (49)
T	total kinetic energy, slug-ft^2/sec^2
U	total potential energy, slug-ft^2/sec^2
$\underline{V}$	velocity of the body-frame origin with respect to the local Earth frame, $= d/dt_I \underline{P} = \dot{\underline{P}}$, ft/sec
α	angle-of-attack, radians
Δ	increment to a variable
$[\Delta J]_j$	first partial derivative of the aircraft inertia matrix with respect to elastic mode j, slug-ft^2, equation (11)
$[\Delta^2 J]_{jk}$	second partial derivative of the aircraft inertia matrix with respect to elastic modes j and k, slug-ft^2, equation (13)
$\underline{\Theta}_i$	rotational deformation at node i, rad
$\underline{\Phi}_{ij}$	component of translational deformation at node i due to elastic mode j, ft

* Research Engineer, Member AIAA

** Project Structures Engineer, Member AIAA

η_j the generalized coordinate cooresponding to elastic mode j, nondimensional

$\underline{\omega}$ vector describing the rotational velocity of the body frame with respect to the local Earth frame, rad/sec

Subscripts

g	due to gravity
i	indexed by the nodes
inert	due to inertial loading
j	indexed by the elastic modes
k	indexed by the elastic modes
s	due to strain energy
o	reference condition or value

Superscripts

as	antisymmetric mode
j	indicial summation over the elastic modes
k	indicial summation over the elastic modes
sy	symmetric mode
T	transpose
^	nondimensional or normalized
'	calculated by different method

Operators

d/dt_B	time rate of change of a vector with respect to an observer in the body frame
d/dt_I	time rate of change of a vector with respect to an observer in the local Earth frame, assumed inertial
d/dt	time rate of change of a scalar
$\dot{x}$	d/dt of the scalar x
$\ddot{x}$	d^2/dt^2 of the scalar x
[]	3×3 matrix
$\underline{x}$	3×1 column vector
$\underline{\overset{\circ}{x}}$	$d/dt_B(\underline{x})$
$\underline{\dot{x}}$	$d/dt_I(\underline{x})$
$\underline{x}\cdot\underline{y}$	scalar (dot) product of 2 vectors, $\underline{x}^T\underline{y}$
$\underline{x}\,\underline{y}^T$	3×3 outer product matrix
$x_j y^j$	indicial sum over the elastic modes
$\underline{x}\times\underline{y}$	cross product of $\underline{x}$ with $\underline{y}$
$\sum_i$	sum over the nodal points, i

Acronyms

EAL	Engineering Analysis Language
FIT	Functional Integration Technology
ISAC	Interaction of Structures, Aerodynamics and Controls
LaRC	Langley Research Center
NASTRAN	NASA Structural Analysis

I. Introduction

In order to realize predicted performance benefits, the design of future aircraft will require multidisciplinary integrated design and analysis.[1] Reference 1 defines "functional integration" as the integration of independently designed subsystems so that adverse interactions are minimized. Examples of such subsystems are the pilot/vehicle interface, flight path/attitude control, structural control, engine control, and weapons systems. The Functional Integration Technology (FIT) team was formed at the Langley Research Center (LaRC) to perform enabling research in the area of functional integration. This paper describes work pertaining to the interaction between structural control and flight path/attitude control. The F/A-18 aircraft was chosen as the focus vehicle because an F/A-18, the proposed High Angle-of-attack Research Vehicle (HARV), is available to NASA at the Dryden Flight Research Facility.

The use of quasi-static-elastic models to predict aircraft response relies on sufficient frequency separation between rigid-body and elastic modes. If this frequency separation does not exist, then the elastic dynamics must be explicitly included in a dynamic model. Flight test programs have demonstrated the feasibility of improving aircraft performance by using feedback control to provide static stability, maneuver load alleviation, and/or increased flutter mode damping.[2] These control design objectives lead to a high-gain, high-bandwidth control strategy that reduces inherent frequency separation between the rigid-body and elastic dynamics and increases the possibility for adverse coupling when control loops are closed.[3] Current aircraft have used leading-edge control devices to maintain roll performance despite wing deformation at high dynamic pressure.[4] Elastic modes such as structural deformation of leading-edge control devices and wing first-bending for a forward-swept-wing configuration tend to destiffen at high dynamic pressure and can thereby interact with lower frequency modes.[5,6] In addition, wing stores reduce elastic vibration frequencies and are subject to centrifugal loads that require flight envelope restriction.[4]

If integrated control of structures and flight path/attitude is to be achieved, then an integrated comprehensive model is required to evaluate candidate designs. The models used for flight path/attitude prediction and real-time simulation are characterized by nonlinear equations of motion suitable for large amplitude maneuvers and nonlinear steady-flow (except for $\dot{\alpha}$ terms) aerodynamic loads. The models used for aeroelastic analysis and design are typically linearized at straight and level flight and are most conveniently defined in the frequency domain. Part of the FIT effort has been to develop a simulation model that integrates the modeling techniques of rigid-body flight path prediction and aeroelastic analysis.

As a first step, the equations of motion of an elastic airplane were examined using Lagrangian mechanics. The airplane is idealized as a collection of lumped masses and lumped inertias being displaced about an accelerated mean reference body frame. Nonlinear inertial coupling terms involving products of rigid-body angular rates, structural deformation, and structural deformation rates were identified. Explanations and detailed analyses of nonlinear inertial coupling in elastic systems can be found in references 7 and 8. The nonlinear terms enable an internal resonance known as

autoparametric coupling that can arise in certain configurations such as an aircraft with wing stores or a T-tail.[7]

The structural model obtained for the F/A-18 contains a tip missile and launcher. During a rapid roll maneuver, differences were found in predicted modal response for an elastic mode dominated by missile movement when the inertial coupling terms are included in the simulation model.

II. Development of Equations

Assumptions

(A1) The aircraft is idealized as a collection of lumped-mass elements, each being a finite rigid body, and each having an associated mass and moments of inertia.
(A2) The elastic restoring force resulting from displacement of any mass element is linear and proportional to that displacement.
(A3) The total rotational displacement of any lumped mass with respect to its undeformed orientation is small.
(A4) Deformation is described by a linear sum of mode shapes multiplied by their time-dependent participation coefficients.
(A5) Gravity is constant over the aircraft.
(A6) The sea-level local Earth frame is assumed to be an inertial frame.

Comments on the Assumptions

The aircraft is regarded as dynamically equivalent to a set of lumped-mass elements. Each mass element resides at a node of a structural finite-element model and constitutes a lumped resistance to acceleration. A mass element should not be confused with the structural finite elements that exist between the nodes. The nature of the mass element is determined by the degrees of freedom allowed at each node of the structural model. In the most general case, each node has six degrees of freedom -- three translational and three rotational. In this case, a mass element located at node i is described by a lumped mass, dm_i, and associated moments of inertia. The 3×3 lumped-inertia matrix, $[dI]_i$, are the mass-element moments of inertia referenced to the aircraft body frame. The mass element is assumed to translate and rotate as a rigid body. The translational deformation at node i from the undeformed reference position is denoted by the vector, $\underline{d}_i$. If the change in orientation of a mass element due to deformation is assumed to be small, then its net rotation with respect to the undeformed position can be represented as a vector, $\underline{\theta}_i$. Treating the rotational deformations as vectors is essential when using a modal approach to describe total deformation. Otherwise, a direction cosine matrix would have to be calculated at each node.

Approach for Equation Development

The effects of both lumped masses and lumped inertias are included in the simulation model. However, for conciseness of presentation, only translational deformations of vehicle mass elements are considered in the development of the equations of motion. Similar treatments can be found in references 3, 9, and 10. The inclusion of rotational deformation is less common in the literature for aircraft, but can be found in reference 11 in the context of connected rigid and deformable bodies. Accounting for the rotational degrees of freedom in deformation adds considerable length to the derivations but little additional insight. Their inclusion parallels the development carried out below and does not change the final form of the equations.

First the total kinetic energy, T, is calculated for the aircraft by summing over all the lumped masses. Using the Rayleigh-Ritz[12] method of assumed modes, kinetic energy is written in terms of modal degrees of freedom by expressing the deformations at each node as a linear sum of participations in the elastic modes. Using the modes of free vibration as assumed modes allows strain potential energy to be expressed as a sum of the modal displacements multiplied by their respective generalized masses and by their natural frequencies of vibration. Although the method of assumed modes is natural for dynamic analysis, it should be noted that since the number of nodes in the structural model is typically much greater than the number of elastic modes retained in an analysis, the use of generalized modal coordinates amounts to reduction of the dynamic model by simple truncation.

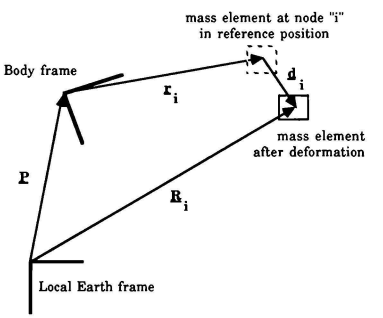


Fig. 1 Vector definitions.

Kinetic Energy

The location of the lumped-mass element in the inertial frame (Fig. 1) is given by

$$\underline{R}_i = \underline{P} + \underline{r}_i + \underline{d}_i \tag{1}$$

The following conventions are used for taking derivatives of vectors with respect to time: a solid dot over a vector denotes the time rate of change with respect to the inertial frame; an open dot denotes the time rate of change with respect to the body frame. A standard kinematic transformation defines the relationship between the time rate-of-change of a vector with respect to a fixed frame and the rate-of-change of the same vector with respect to a rotating frame. Thus,

$$\dot{\underline{x}} = \mathring{\underline{x}} + \underline{\omega} \times \underline{x} \tag{2}$$

Taking the time derivative of (1) according to (2) and recognizing that $\mathring{\underline{r}}_i = \underline{0}$, we obtain for the inertial velocity of the lumped mass, dm_i

$$\underline{v}_i \triangleq \dot{\underline{R}}_i = \underline{V} + \underline{\omega}\times(\underline{r}_i+\underline{d}_i) + \mathring{\underline{d}}_i \tag{3}$$

where we have denoted $\dot{\underline{P}}$, the velocity of the body-frame origin in the inertial frame, as $\underline{V}$. The kinetic energy for the total aircraft is then

$$T = \frac{1}{2}(\sum_i \underline{v}_i\cdot\underline{v}_i dm_i) \tag{4}$$

where the symbol $\sum_i$ refers to a summation over all the lumped masses. Carrying out the expansion indicated by (3) results in

$$\begin{aligned} T = {} & \frac{1}{2}\underline{V}\cdot\underline{V}(\sum_i dm_i) + \frac{1}{2}\sum_i(\mathring{\underline{d}}_i\cdot\mathring{\underline{d}}_i)dm_i \\ & + \frac{1}{2}\underline{\omega}^T[\sum_i\{(\underline{r}_i\cdot\underline{r}_i)[I] - \underline{r}_i\underline{r}_i^T\}dm_i]\underline{\omega} \\ & + \frac{1}{2}\underline{\omega}^T[\sum_i\{(\underline{d}_i\cdot\underline{r}_i+\underline{r}_i\cdot\underline{d}_i)[I] - \underline{d}_i\underline{r}_i^T - \underline{r}_i\underline{d}_i^T\}dm_i]\underline{\omega} \\ & + \frac{1}{2}\underline{\omega}^T[\sum_i\{(\underline{d}_i\cdot\underline{d}_i)[I] - \underline{d}_i\underline{d}_i^T\}dm_i]\underline{\omega} \\ & + \underline{\omega}\cdot\{\sum_i(\underline{d}_i\times\mathring{\underline{d}}_i)dm_i\} \\ & + \underline{V}\cdot\{\underline{\omega}\times(\sum_i\underline{r}_i dm_i)\} + \underline{V}\cdot\{\underline{\omega}\times(\sum_i\underline{d}_i dm_i)\} \\ & + \underline{V}\cdot\{\sum_i\mathring{\underline{d}}_i dm_i\} + \underline{\omega}\cdot\{\sum_i(\underline{r}_i\times\mathring{\underline{d}}_i)dm_i\} \end{aligned} \tag{5}$$

The significance of each of the terms in (5) is described in the following discussion.

Kinetic Energy in Modal Components

The translational deformation at node i, $\underline{d}_i$, is considered to be composed of a finite sum of mode shapes, $\underline{\Phi}_{ij}$, and their time-dependent participation coefficients, η_j. Thus

$$\underline{d}_i = \underline{\Phi}_{ij}\eta^j \tag{6}$$

where the notation $()_j()^j$ represents an indicial sum over the elastic modes. In this paper the elastic modes are always indexed (superscripts and subscripts) by j or k. For the time rate-of-change with respect to the body frame we have

$$\mathring{\underline{d}}_i = \underline{\Phi}_{ij}\dot{\eta}^j \tag{7}$$

Note $\dot{\eta}_j$(or $\dot{\eta}^j$) is an unambiguous time derivative since η_j is a scalar. The expression for kinetic energy in (5) can be put in modal form by using equations (6) and (7). The resulting modal inertial quantities are discussed below.

The total mass of the aircraft is

$$\sum_i dm_i = m \tag{8}$$

The second term of (5) becomes

$$\begin{aligned} \sum_i(\mathring{\underline{d}}_i\cdot\mathring{\underline{d}}_i)dm_i &= \sum_i(\underline{\Phi}_{ij}\dot{\eta}^j)\cdot(\underline{\Phi}_{ik}\dot{\eta}^k)dm_i \\ &= \{\sum_i\underline{\Phi}_{ij}\cdot\underline{\Phi}_{ik}dm_i\}\dot{\eta}^j\dot{\eta}^k = M_{jk}\dot{\eta}^j\dot{\eta}^k \end{aligned} \tag{9}$$

where M_{jk} is the element of the generalized modal mass matrix corresponding to modes j and k.

The rotational inertia matrix for the total undeformed aircraft is defined from the third term of equation (5) as

$$\sum_i((\underline{r}_i\cdot\underline{r}_i)[I] - \underline{r}_i\underline{r}_i^T)dm_i = [J_o] \tag{10}$$

The first-order effect of deformation on the total time-varying vehicle inertia matrix arises from the fourth term of (5). In modal form, it becomes the first partial derivative of [J] with respect to modal participation. Thus,

$$\begin{aligned} \sum_i\{(\underline{d}_i\cdot\underline{r}_i + \underline{r}_i\cdot\underline{d}_i)[I] - \underline{d}_i\underline{r}_i^T - \underline{r}_i\underline{d}_i^T\}dm_i & \\ = \{\sum_i((2\underline{r}_i\cdot\underline{\Phi}_{ij})[I] - \underline{\Phi}_{ij}\underline{r}_i^T - \underline{r}_i\underline{\Phi}_{ij}^T)dm_i\}\eta^j & \\ = [\Delta J]_j\eta^j \quad \text{,where } [\Delta J]_j = [\Delta J]_j^T & \end{aligned} \tag{11}$$

The second-order effect of deformation on the total vehicle inertia matrix can be derived from the fifth term of (5) as

$$\begin{aligned} \sum_i\{(\underline{d}_i\cdot\underline{d}_i)[I] - \underline{d}_i\underline{d}_i^T\}dm_i & \\ = \{\sum_i((\underline{\Phi}_{ij}\cdot\underline{\Phi}_{ik})[I] - \underline{\Phi}_{ij}\underline{\Phi}_{ik}^T)dm_i\}\eta^j\eta^k & \\ = [A]_{jk}\eta^j\eta^k \quad \text{,where } [A]_{jk}^T = [A]_{kj} & \end{aligned} \tag{12}$$

The second partial derivative of [J] with respect to modal participation in modes j and k can be defined by

$$[\Delta^2 J]_{jk} = [A]_{jk}+[A]_{kj} \tag{13}$$

Second-order momentum coupling between the elastic modes and angular momentum arises from summing the cross products of modes j and k over the aircraft. The sixth term of equation (5) can be written as

$$\sum_i(\underline{d}_i\times\mathring{\underline{d}}_i)dm_i = \{\sum_i(\underline{\Phi}_{ij}\times\underline{\Phi}_{ik})dm_i\}\eta^j\dot{\eta}^k = \underline{h}_{jk}\eta^j\dot{\eta}^k \tag{14}$$

Choosing the origin of the body frame to be at the center-of-mass of the aircraft in the undeformed reference condition removes the seventh term in equation (5) since

$$\sum_i\underline{r}_i dm_i = \underline{0} \tag{15}$$

Using the terms defined in equations (6)-(15) to reduce equation (5), we get the following total kinetic energy expression for arbitrary deformation about the body reference frame:

$$T = \frac{1}{2}m\underline{V}\cdot\underline{V} + \frac{1}{2}\underline{\omega}^T[J]\underline{\omega} + \frac{1}{2}M_{jk}\dot{\eta}^j\dot{\eta}^k + \underline{\omega}\cdot\underline{h}_{jk}\eta^j\dot{\eta}^k + \Delta T \tag{16}$$

where,

$$[J] = [J_o] + [\Delta J]_j\eta^j + \frac{1}{2}[\Delta^2 J]_{jk}\eta^j\eta^k \tag{17}$$

and,

$$\Delta T = \underline{V}\cdot\underline{\omega}\times\{\sum_i \underline{\Phi}_{ij} dm_i\}\eta^j + \underline{V}\cdot\{\sum_i \underline{\Phi}_{ij} dm_i\}\dot{\eta}^j + \underline{\omega}\cdot\{\sum_i \underline{r}_i\times\underline{\Phi}_{ij} dm_i\}\dot{\eta}^j \quad (18)$$

Constraints on Deformation

If the assumed mode shapes, $\underline{\Phi}_{ij}$, are the free vibration eigenvectors of a restrained structure, the bracketed modal quantities in (18) will in general not be zero and will have to be retained in (16). A typical restraint is to constrain nodal deflection and slope at one point of the structure. The purpose of the restraint is to make the stiffness matrix invertible. The boundary conditions for undamped vibration of an unrestrained structure, however, are defined in modal form as follows:

$$\sum_i \underline{\Phi}_{ij} dm_i = \underline{0} \quad \text{and} \quad \sum_i \underline{r}_i\times\underline{\Phi}_{ij} dm_i = \underline{0} \quad (19)$$

Equation (19) represents, in modal form, constraints that are called "practical" mean-axes constraints in reference 3, "approximate mean reference frame conditions" in reference 10, and conditions for a Buckens floating reference frame in reference 11. Dusto, et al. define a "mean reference frame" as a frame for which an observer, moving with the frame, would observe the kinetic energy of the relative elastic motion to be a minimum.[10] Equation (19) is a linearized modal form of the constraints that minimize relative kinetic energy. If the assumed mode shapes are the eigenvectors of a structural model in undamped vibration with free-free boundary conditions, then motion according to each mode shape j should satisfy the conditions of equation (19). In this case ΔT is zero and we have for the total kinetic energy in modal coordinates

$$T = \tfrac{1}{2} m \underline{V}^T \underline{V} + \tfrac{1}{2}\underline{\omega}^T [J]\underline{\omega} + \tfrac{1}{2} M_{jk}\dot{\eta}^j\dot{\eta}^k + \underline{\omega}^T \underline{h}_{jk}\eta^j\dot{\eta}^k \quad (20)$$

Potential Energy

The potential energy, U, of the airplane is composed of a gravity component and a component due to strain energy. The gravity component can be approximated by calculating the work done by gravity in bringing each mass element to its geometric altitude and summing over the mass elements. Applying assumption (A5) and making use of equations (15) and (19) we obtain, as in reference 3

$$U_g = - m\underline{g}\cdot\underline{P} \quad (21)$$

Observe that $\underline{g}\cdot\underline{P}$ is typically negative. Since the modes of undamped free vibration are being used as generalized coordinates, the strain energy due to modal participation is given by[12]

$$U_s = \tfrac{1}{2} M_{jj}\omega_j^2(\eta^j)^2 \quad (22)$$

where ω_j is the undamped natural frequency of elastic mode j. The total potential energy is then

$$U = U_g + U_s = - m\underline{g}\cdot\underline{P} + \tfrac{1}{2} M_{jj}\omega_j^2(\eta^j)^2 \quad (23)$$

Lagrange's Equations

Forming the Lagrangian, T - U, we can form a vector equation for translational momentum, a vector equation for angular momentum, and a scalar equation for each elastic mode. Thus,

$$\frac{d}{dt_B}\left\{\frac{\partial(T-U)}{\partial \underline{V}}\right\} + \underline{\omega}\times\left\{\frac{\partial(T-U)}{\partial \underline{V}}\right\} - \partial(T-U)/\partial\left(\int \underline{V} dt\right) = \underline{F} \quad (24)$$

$$\frac{d}{dt_B}\left\{\frac{\partial(T-U)}{\partial \underline{\omega}}\right\} + \underline{\omega}\times\left\{\frac{\partial(T-U)}{\partial \underline{\omega}}\right\} - \partial(T-U)/\partial\left(\int \underline{\omega} dt\right) = \underline{L} \quad (25)$$

$$\frac{d}{dt}\{\partial(T-U)/\partial\dot{\eta}_j\} - \partial(T-U)/\partial\eta_j = Q_{\eta_j} \quad (26)$$

Translational Momentum

The components of (24) are expanded below as

$$\begin{aligned} &\partial T/\partial \underline{V} = m\,\underline{V} \\ &\partial U/\partial \underline{V} = \underline{0} \\ &\partial T/\partial\left(\int \underline{V} dt\right) = \underline{0} \\ &\partial U/\partial\left(\int \underline{V} dt\right) = \partial(\dot{U})/\partial(\underline{V}) \end{aligned} \quad (27)$$

Since the gravity vector at any point in space is constant with respect to a local Earth frame that has been assumed to be inertial, we have for the time rate-of-change of potential energy

$$\begin{aligned} \dot{U} &= - m\dot{\underline{g}}\cdot\underline{P} - m\underline{g}\cdot\dot{\underline{P}} + M_{jj}\omega_j^2\eta^j\dot{\eta}^j \\ &= - m\underline{g}\cdot\underline{V} + M_{jj}\omega_j^2\eta^j\dot{\eta}^j \end{aligned} \quad (28)$$

Taking the partial derivative of the scalar $\dot{U}$ with respect to the vector $\underline{V}$ results in

$$\partial(\dot{U})/\partial(\underline{V}) = - m\underline{g} \quad (29)$$

Equation (29) is the standard gravity contribution to the translational momentum equations. Applying (27) and (29) to (24) produces

$$\frac{d}{dt_B}(m\underline{V}) + \underline{\omega}\times(m\underline{V}) = \underline{F} + m\underline{g} \quad (30)$$

Clearing the left side of (30) of all but the acceleration terms gives the desired translational momentum equation. Thus,

$$m\dot{\underline{V}} = \underline{F} - m(\underline{\omega}\times\underline{V}) + m\underline{g} \quad (31)$$

Note the absence of inertial coupling between translational and elastic momentum that results from the linearized mean-axes constraints.

Angular Momentum

The components of equation (25) are

$$\begin{aligned} &\partial T/\partial \underline{\omega} = [J]\underline{\omega} + \underline{h}_{jk}\eta^j\dot{\eta}^k \\ &\partial U/\partial \underline{\omega} = \underline{0} \\ &\partial T/\partial\left(\int \underline{\omega} dt\right) = \underline{0} \\ &\partial U/\partial\left(\int \underline{\omega} dt\right) = \underline{0} \end{aligned} \quad (32)$$

from which we can calculate

$$\frac{d}{dt_B}(\partial T/\partial \underline{\omega}) = [J]\mathring{\underline{\omega}} + [\mathring{J}]\underline{\omega} + \underline{h}_{jk}\dot{\eta}^j\dot{\eta}^k + \underline{h}_{jk}\eta^j\ddot{\eta}^k \quad (33)$$

and

$$\underline{\omega}\times(\partial T/\partial \underline{\omega}) = \underline{\omega}\times[J]\underline{\omega} + \underline{\omega}\times\underline{h}_{jk}\eta^j\dot{\eta}^k \quad (34)$$

Applying (32) through (34) to (25) yields

$$[J]\mathring{\underline{\omega}} + [\mathring{J}]\underline{\omega} + \underline{h}_{jk}\dot{\eta}^j\dot{\eta}^k + \underline{h}_{jk}\eta^j\ddot{\eta}^k + \underline{\omega}\times[J]\underline{\omega} + \underline{\omega}\times\underline{h}_{jk}\eta^j\dot{\eta}^k = \underline{L} \quad (35)$$

Rearranging (35) so that only acceleration terms appear on the left side results in the angular momentum equation in modal form. Thus,

$$[J]\mathring{\underline{\omega}} + \underline{h}_{jk}\eta^j\ddot{\eta}^k = \underline{L} - \underline{\omega}\times[J]\underline{\omega} - [\mathring{J}]\underline{\omega} - \underline{h}_{jk}\dot{\eta}^j\dot{\eta}^k - \underline{\omega}\times\underline{h}_{jk}\eta^j\dot{\eta}^k \quad (36)$$

Using the expression for the time-varying inertia matrix given in (17), the time rate-of-change of the total inertia matrix in the body frame can be expressed as follows:

$$[\mathring{J}] = [\Delta J]_j\dot{\eta}^j + [\Delta^2 J]_{jk}\eta^j\dot{\eta}^k \quad (37)$$

Elastic Mode Degrees of Freedom

The components of equation (26) are

$$\begin{aligned}
\partial T/\partial\dot{\eta}_j &= M_{jk}\dot{\eta}^k + \underline{\omega}\cdot\underline{h}_{kj}\eta^k \\
\partial U/\partial\dot{\eta}_j &= \underline{0} \\
\partial T/\partial\eta_j &= \tfrac{1}{2}\,\underline{\omega}^T[[\Delta J]_j + [\Delta^2 J]_{jk}\eta^k]\underline{\omega} + \underline{\omega}\cdot\underline{h}_{jk}\dot{\eta}^k \\
\partial U/\partial\eta_j &= M_{jj}\omega_j^2\eta_j
\end{aligned} \quad (38)$$

from which one obtains

$$-\frac{d}{dt}(M_{jk}\dot{\eta}^k + \underline{\omega}\cdot\underline{h}_{kj}\eta^k) - \tfrac{1}{2}\,\underline{\omega}^T[[\Delta J]_j + [\Delta^2 J]_{jk}\eta^k]\underline{\omega} - \underline{\omega}\cdot\underline{h}_{jk}\dot{\eta}^k + M_{jj}\omega_j^2\eta_j = Q_{\eta_j} \quad (39)$$

Expanding the derivative term in (39) yields

$$-\frac{d}{dt}(M_{jk}\dot{\eta}^k + \underline{\omega}\cdot\underline{h}_{kj}\eta^k) = M_{jk}\ddot{\eta}^k + \mathring{\underline{\omega}}\cdot\underline{h}_{kj}\eta^k + \underline{\omega}\cdot\underline{h}_{kj}\dot{\eta}^k \quad (40)$$

Combining (39) and (40) and observing that $\underline{h}_{kj} = -\underline{h}_{jk}$, the equation for elastic mode j can be written as

$$M_{jk}\ddot{\eta}^k - \mathring{\underline{\omega}}\cdot\underline{h}_{jk}\eta^k = Q_{\eta_j} - M_{jj}\omega_j^2\eta_j + 2\underline{\omega}\cdot\underline{h}_{jk}\dot{\eta}^k + \tfrac{1}{2}\,\underline{\omega}^T[[\Delta J]_j + [\Delta^2 J]_{jk}\eta^k]\underline{\omega} \quad (41)$$

In the elastic mode equation above, the term $\mathring{\underline{\omega}}\cdot\underline{h}_{kj}\eta^k$ is due to the angular acceleration of the body frame, $2\underline{\omega}\cdot\underline{h}_{jk}\dot{\eta}^k$ is a Coriolis term, and $\frac{1}{2}\,\underline{\omega}\cdot[[\Delta J]_j + [\Delta^2 J]_{jk}\eta^k]\underline{\omega}$ represents centrifugal loading on the elastic mode.

III. Remarks on Equations

Equations (31),(36), and (41) define the simulation model used in this study. Angular momentum and the elastic mode equations, (36) and (41) respectively, are coupled by two sources: a) modal-induced changes in mass distribution, described in the terms $[\Delta J]_j$ and $[\Delta^2 J]_{jk}$; and b) a second-order coupling term, $\underline{h}_{jk}$, that arises from the presence of modes that do not all act in the same plane, such as in-plane/lateral (fore/aft) bending and out-of-plane/vertical bending of a wing beam.

Comparison with Inertially Decoupled Equations

In the literature for aircraft, the equations of motion that are frequently used for performing stability analyses are as follows:

$$m\mathring{\underline{V}} = \underline{F} - m(\underline{\omega}\times\underline{V}) + m\underline{g} \quad (31)$$

$$[J_o]\mathring{\underline{\omega}} = \underline{L} - \underline{\omega}\times[J_o]\underline{\omega} \quad \text{from (36)} \quad (42)$$

$$M_{jj}\ddot{\eta}_j = Q_{\eta_j} - M_{jj}\omega_j^2\eta_j \quad \text{from (41)} \quad (43)$$

These equations can be derived by augmenting (A1) through (A6) with the following two assumptions:

(A7) $\underline{\omega}$ is small enough that product terms involving $\underline{\omega}$ and deformation as well as $\underline{\omega}$ and deformation rates can be neglected, in which case the $[\Delta J]$ terms vanish.

(A8) Deformation and deformation rates are parallel, so that the cross-product term, $\underline{h}_{jk}$, vanishes.

Note that there is no inertial coupling between the rigid and elastic modes in (42) and (43). The only coupling is through the applied loads, Q_{η_j}, $\underline{L}$ and $\underline{F}$. Thus, there is no mechanism by which rigid-body rotational rates can inertially load the elastic modes. Equations (36) and (41) are an attempt to extend equations (42) and (43) to the case $|\underline{\omega}| \geq 1$. This issue was raised by Cavin and Dusto, who asserted that an "area requiring further study is that of developing viable procedures for determining when product terms involving {d} and {$\underline{\omega}$} should be included and of defining effective procedures for accomplishing the inclusion".[9]

Extracting stability derivatives from equations (42) and (43) or from (36) and (41), produce, to four or five significant figures, identical answers in unaccelerated flight. The only differences result from nonzero trim values of modal deflections. These deflections affect [J] as indicated in equation 17 and thereby change angular acceleration sensitivities. The terms $[\Delta J]_j$ and $[\Delta^2 J]_{jk}$ were calculated from structural models for both the X-29 and an F/A-18 with tip missiles, and as one would expect, they are small with respect to $[J_o]$. The direct effect of a deformation-induced change in the inertia matrix [J] on rigid-body angular response should almost always be negligible, unless an aircraft that retracts its wings during a roll is being modeled. The fact remains, the terms $[\Delta J]_j$ and $[\Delta^2 J]_{jk}$ are, along with the Coriolis term, the only way steady rotation rates can affect modal response in the elastic degree of freedom equation. While the values for elements of $[\Delta J]_j$

and $[\Delta^2 J]_{jk}$ are small with respect to $[J_o]$, they can be an order of magnitude larger than generalized modal masses.

Effect of the Lumped Inertias

Although it was not indicated in the equations of motion development, nodal rotational deformation was included in the mathematical model for completeness and because the lumped inertias in the beam-element structural model accounted for 10 percent of the total x-axis moment of inertia in the undeformed condition. The inertia matrix for the F/A-18 aircraft in the undeformed reference condition, $[J_o]$, is shown below. The moments and products of inertia are calculated from the structural model that includes tip missiles.

The model total reference inertia matrix with lumped inertias is shown below without derivation.

$$[J_o]' = \sum_i \left| ((\underline{r}_i^T \underline{r}_i)[I] - \underline{r}_i \underline{r}_i^T) dm_i + [dI]_i \right| \qquad (44)$$

$$= \begin{vmatrix} 24390. & 0. & -1450. \\ 0. & 130600. & 0. \\ -1450. & 0. & 152200. \end{vmatrix} \text{ (slug-ft}^2\text{)}$$

The inertia matrix due to lumped masses alone is

$$[J_o] = \sum_i ((\underline{r}_i^T \underline{r}_i)[I] - \underline{r}_i \underline{r}_i^T) dm_i \qquad (10)$$

$$= \begin{vmatrix} 22000. & 0. & -1450. \\ 0. & 126400. & 0. \\ -1450. & 0. & 147700. \end{vmatrix} \text{ (slug-ft}^2\text{)}$$

In including rotational components of deformation, one observes that in the angular momentum equation, (36), and in the equation for the elastic modes, (41), each of the inertial terms has two components, one arising from the lumped masses, dm_i, and the other arising from the lumped inertias, $[dI]_i$. The affected terms are $[J_o]$, $[\Delta J]_j$, $[\Delta^2 J]_{jk}$, and $\underline{h}_{jk}$. In addition, the linearized mean-axes constraints of (19) have a $[dI]_i$ term in the rotational constraint. The effect of inclusion of the lumped inertias on $[\Delta J]_j$, for example, is restricted to the off-diagonal components, but can induce sign changes.

IV. Structural Model

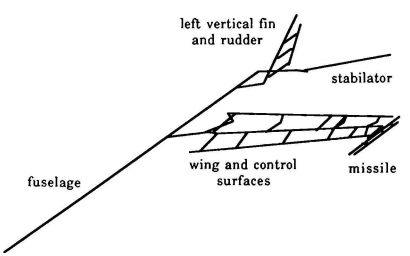


Fig. 2 Beam-element half model of F/A-18.

A NASTRAN beam-element half-model (Fig. 2) of the F/A-18 was obtained from McDonnell Aircraft Company and translated into EAL (Engineering Analysis Language).[13] The model was analyzed for free-free vibration modes with both symmetric and antisymmetric boundary conditions and for internal modal load coefficients. The mode shapes were analyzed for satisfaction of the first-order mean-axes conditions as described in equation (19). Though the free vibration eigen-problem was solved for free-free boundary conditions, the modes did not satisfy the first-order mean-axes conditions to machine accuracy.

Small translational and rotational corrections were computed and applied to the mode shapes. These corrections merely reoriented the modes in space to improve their satisfaction of the first-order mean-axes conditions, but preserved the actual shapes of the modes. Thus, the load coefficients computed within EAL remained valid. Using these corrected mode shapes, the inertial interaction quantities and generalized masses required by the equations of motion were computed.

V. Aerodynamic Model

The steady-flow nonlinear aerodynamic database used by the real-time facility at LaRC was obtained for use in the FIT simulation[14]. The database has flex/rigid multipliers and increments that correct the database for quasi-steady deformation. These correction factors are functions of altitude and/or dynamic pressure. The database corresponding to the undeformed aircraft was recovered by setting the flex/rigid corrections to their limiting values at high altitude or zero dynamic pressure as appropriate.

The same corrected mode shapes used to generate the required inertial quantities are input to the ISAC[15] programs to compute generalized unsteady aerodynamic loads using doublet-lattice theory for a range of reduced frequencies and Mach numbers. For each Mach number, a rational function approximation for the transfer function of the unsteady aerodynamic loads is determined by a least-squares fit to a table of oscillatory loads at various reduced frequencies and for a selected set of aerodynamic "lags"[16].

The aerodynamic loads data are combined as follows. Total forces and moments, $\underline{F}$ and $\underline{L}$, due to rigid-body velocities and control surface positions are calculated from the nonlinear database that represents the undeformed aircraft. Loads at each time step are calculated by linear interpolation of the independent variables. Rigid-body total forces and moments due to deformations, deformation rates, control surface rates and control surface accelerations are calculated from the ISAC-generated database by interpolation of Mach number at each time step. Generalized forces on the elastic modes, Q_{n_j}, are also calculated from the ISAC-generated database.

VI. Model Verification and Validation

Simulation model verification is in progress. Results to date have provided confidence that major errors do not exist. The model was compared with

the LaRC real-time simulation by replicating trim solutions and eigenvalues/vectors at several operating conditions. Future plans include comparisons with flight data generated at NASA Dryden and more extensive comparisons with published flutter analyses. Some preliminary flutter comparisons are described below.

An aeroelastic analysis of the F/A-18 aircraft using the V-g method is documented in reference 5. The V-g method typically varies velocity and computes the additional structural damping, g, required to maintain a harmonic solution in each mode. A flutter mode occurs where g becomes positive for any mode. Yurkovich[5] describes an "8-hertz" flutter mode involving coupling of wing first-torsion with wing second-bending. The "8-hertz" mode occurs at the following conditions: M=.9; 805 knots equivalent airspeed; symmetric analysis with rigid-body pitch and plunge, and 13 elastic modes included. A beam-element structural model and doublet-lattice aerodynamics were used.

The FIT simulation is currently configured for 10 symmetric and 10 antisymmetric elastic modes. A symmetric-only flutter solution was calculated at M=.9 by varying the speed of sound at sea level until neutral stability was reached. At an equivalent airspeed of 650 knots, an eigenvalue with a natural frequency of 9.2 hertz and slightly negative damping was observed. The associated eigenvector was dominated by the velocities of the wing first-torsion, fuselage/stabilator first-bending, and wing second-bending elastic modes.

The trend in Yurkovich's analysis of the "8-hertz" mode indicated that at the 13 elastic mode analysis point, there was about a 25-knot decrease in flutter equivalent airspeed for each elastic mode removed. The fact that a similar flutter mechanism was observed at about the same frequency as indicated by Yurkovich[5] was interpreted as a qualified confirmation of the FIT simulation.

VII. Results

An attempt was made to address the question of where in the flight envelope and for what feasible maneuvers the nonlinear inertial-coupling terms, $[\Delta J]_j$, $[\Delta^2 J]_{jk}$, and $\underline{h}_{jk}$, have measurable impact on either rigid-body or elastic response. A number of time histories were generated with and without the nonlinear coupling terms included in the model. The range of maneuvers was limited since a closed-loop simulation was not available. Large-amplitude control doublets in the pitch, roll, and yaw axes were considered in addition to trims in coordinated turns. The altitudes examined were sea level and 30,000 feet. Use of doublet-lattice unsteady aerodynamics dictated that Mach number be subsonic. A viscous damping term with a damping ratio of .005 was included in each elastic mode equation.

In no case were significant differences observed when comparing rigid-body responses of the inertially coupled model to the inertially decoupled model. In general, pitch and yaw maneuvers did not generate sufficient angular rates to excite noticeable differences in elastic response for an aircraft as stiff as the modeled F/A-18. In addition, rapid pitch rates quickly drive the model out of the low-incidence regime where doublet-lattice theory is valid. However, in contrast to pitch and yaw rates, attainable roll rates are of a magnitude sufficient to affect elastic response. A case that illustrates the effect of the inertial coupling terms in elastic response is described below.

The open-loop simulation was trimmed straight and level at Mach = .7 and an equivalent airspeed of 781 ft/sec. A 20-degree doublet was applied to commanded differential ailerons at the actuator inputs. A stabilator/aileron interconnect with a gain of .5 degree/degree was employed to provide a corresponding input to the stabilator actuators. This combined command was used in order to create higher roll rates. All other control surfaces were held fixed, producing a somewhat uncoordinated roll.

It should be noted that the roll rates generated by the simulation probably overestimate the performance of the aircraft. Loss of aileron effectiveness is currently underpredicted in the FIT aeroelastic model. Underprediction of F/A-18 aileron effectiveness loss is also described in reference 4.

Results indicated by the solid lines in figures 3 through 6 correspond to the model with inertial coupling terms included (terms on) according to equations (31), (36), and (41). Responses indicated by the dashed lines are for the model with inertial coupling terms removed (terms off), as defined by equations (31), (42), and (43). Where there are no dashed lines, the results for the two models agreed. The modal coordinates shown in the figures have been scaled. This scaling is such that a unit deflection of each mode has the same strain energy as a unit deflection in the lowest frequency mode (the first symmetric mode). This scaling is described later.

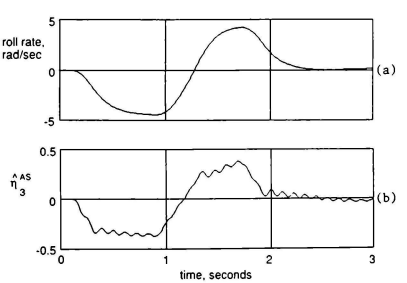


Fig. 3 Lateral/antisymmetric response.

Figures 3.a and 3.b indicate the lateral/antisymmetric response of both models. The rigid-body response is conventional and the inertial coupling terms made no significant difference in any antisymmetric elastic response. Figure 3.b is typical of these elastic responses. It shows the third antisymmetric elastic mode, which is characterized by wing first-torsion and missile pitch.

This mode "rings" near its undamped frequency of 8.9 hertz.

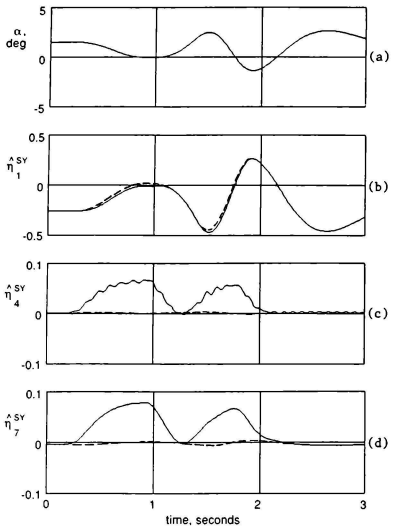


Fig. 4 Longitudinal/symmetric response.

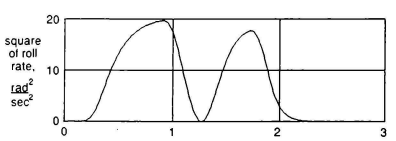


Fig. 5 Response of roll rate squared.

Figure 4 shows the longitudinal/symmetric response of the models. It is here that the coupling terms have noticeable effect on elastic response. Figure 4.b is the response of the first symmetric mode (wing first-bending). Its loading is almost entirely aerodynamic and it closely tracks the angle-of-attack trace in figure 4.a. A small amount of inertial loading is indicated by the difference in the dashed and solid lines. Figure 4.c shows the response of the fourth symmetric mode, a mode characterized by missile yaw and fin bending. Figure 4.d is the seventh symmetric mode's response, a wing in-plane/lateral (fore/aft) first-bending mode. Figure 5 is the response of p^2, which is the principal cause of the inertial loadings observed in figures 4.c and 4.d. Both modes experience almost no aerodynamic loading, as can be observed by the lack of influence of angle-of-attack on their responses and their near-zero trim deflections. The 12.2-hertz natural frequency of the fourth symmetric mode is easily seen being excited by the nonlinear inertial coupling terms.

The inertially-induced increment in the response of the fourth and seventh symmetric modes can be predicted if equation (41) is solved for steady-state modal displacement by assuming constant $\underline{\omega}$. Deleting terms with derivatives of η_j and $\underline{\omega}$, and supposing that $[\Delta^2 J]_{jk}\eta^k << M_{jj}\omega_j^2$, one gets

$$M_{jj}\omega_j^2\eta_j = Q_{\eta_j} + \frac{1}{2}\,\underline{\omega}^*[\Delta J]_j\underline{\omega} \qquad (45)$$

Solving the quasi-steady equation (45) for η_j, the inertial component of the quasi-steady effect is

$$\Delta\eta_{j_{inert}} = \frac{1}{2}\,\underline{\omega}^*\{[\Delta J]_j M_{jj}^{-1}\}\underline{\omega}\,/\omega_j^2 \qquad (46)$$

The modal coordinates in (46) are normalized to a 1-foot deflection at the point of maximum deflection. The question of how to scale eigenvectors to convey physical insight is problem dependent. In reference 17, a handling qualities study, the modal coordinates were scaled according to mode slope at the pilot station. For this analysis, the modal coordinates are scaled so that a unit deflection in any mode results in the same strain energy. This can be accomplished by

$$\hat{\eta}_j = \{M_{jj}/M_{11}^{SY}\}^{1/2}\,\{\omega_j/\omega_1^{SY}\} \qquad (47)$$

Combining (46) and (47) gives the scaled quasi-steady elastic response to steady angular velocity as

$$\Delta\hat{\eta}_{j_{inert}} = \frac{1}{2}\{M_{jj}/M_{11}^{SY}\}^{1/2}\,\underline{\omega}^*\{[\Delta J]_j M_{jj}^{-1}\}\underline{\omega}/\{\omega_j\omega_1^{SY}\} \qquad (48)$$

A scalar-valued function of $\underline{\omega}$ defined by

$$R_j(\underline{\omega}) = \Delta\hat{\eta}_{j_{inert}} \qquad (49)$$

can then be interpreted as parameter reflecting the inertial influence on each mode of a given value of $\underline{\omega}$. One logical choice for $\underline{\omega}$ would be maximum design values. The parameter, R_j, is easily calculated for a given set of vibration modes. If any mode were singled out by its R_j value, it would then remain to determine the influence of that mode on either the other modes through aerodynamic influence coefficients, or on physical figures-of-merit such as selected loads or accelerations. Thus a two-part test for the inclusion of inertial coupling is suggested. The first part determines inertially affected modes by their relatively large values of R_j. The second and more difficult part would assess the importance of inertially affected modes upon dynamic characteristics of interest.

Table 1 shows R_j calculated for the 20 elastic modes used in the FIT simulation. The $\underline{\omega}$ used in the calculations corresponds to the peak roll rate that occurs at about .9 seconds (Fig. 5.a). The

R_j's should approximate the average increment between "terms on" and "terms off" in the $\hat{n}_j$ responses shown in figure 4. The modal inertial data required for the calculations are given in table 2. An explanation for the lack of observed response to inertial loading in the antisymmetric modes is found in the form of the scaled $[\Delta J]^{sy}_j$ matrices in table 2. Due to zero diagonals for the $[\Delta J]^{as}_j$, the antisymmetric modes are "forced" by the cross products, pq and qr, which are much smaller than p^2 at the peak roll-rate point. The most affected antisymmetric mode is the second mode and involves lateral fuselage bending and missile yaw.

Mode	Symmetric R_j	Antisymmetric R_j
1	-.0283	.0008
2	.0048	-.0120
3	.0064	.0004
4	.0739	.0039
5	.0075	-.0001
6	-.0028	-.0090
7	.0899	-.0016
8	-.0115	-.0006
9	-.0043	.0018
10	-.0027	-.0012

$\underline{\omega}^T$ = [4.74,.116,.025] rad/sec

Table 1. Modal sensitivity to $\underline{\omega}$

For the simulation model, the modes that are most affected by the nonlinear coupling terms, symmetric modes four and seven, are also modes that are otherwise virtually decoupled from the rest of the model. Because these modes predominantly involve in-plane bending of the primary lifting surface, and doublet-lattice theory does not predict in-plane loads, there is no aerodynamic coupling with the other symmetric modes. Unless calculation of distributed drag loads is included in the model, these modes will remain decoupled. So while the parameter R_j flags the fourth and seventh symmetric modes as inertially affected modes, the second part of the test cannot be applied in this case since these modes are not coupled to modeled dynamics.

In a situation where massive stores were suspended below the wing, conventional beam bending of the attachment structure would cause store elastic displacement to occur in the aircraft y-axis and directly impact the $\underline{r}^T\underline{d}$ terms used to calculate $[\Delta J]_j$. If coupled with wing out-of-plane bending, the store displacement described above could be expected to occur at relatively low frequency. Furthermore, wing out-of-plane bending would aerodynamically impact the rest of the model. However, it has been observed that the trend in fighter design is toward conformal stores. Therefore, the possibility of stores hanging under the wings of future aircraft designs is unlikely.

Dynamic loads were calculated using internal modal load coefficients generated by EAL. Load stations considered were stabilator root, fin root, missile attachment points, and wing root. The out-of-plane/vertical shear load at the left wing root is shown in figure 6.a. There is total agreement between the terms on/off cases, in spite of perceptible differences in the response of the first symmetric mode (wing first-bending) shown in figure 4.b. The load response is aerodynamically dominated and closely tracks both α and the first symmetric mode. The trim loading is about 10,000 lbs on each wing. Since the aircraft at trim weighed 33,300 lbs and the fuselage and tail are treated as lifting surfaces, this number is reasonable. The only loads for which the inertial coupling terms made any difference were in the fore/aft direction, which is the principle displacement direction of symmetric mode seven. Fore/aft shear load at the left wing-root is shown in figure 6.b. While the magnitude difference is substantial, it is not at all clear whether this is a good prediction or the result of a failure of the loads calculation procedure to converge to the correct answer. The loads calculation method is a variant of the "mode displacement method"[10] which is known to suffer convergence difficulties when too few modes are used to define deformation. If more (i.e. higher frequency) fore/aft modes were added, one would expect the dashed line to approach zero, reflecting the lack of modeled in-plane aerodynamic loading.

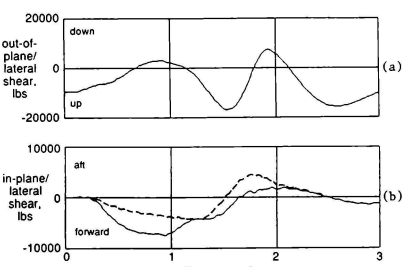


Fig. 6 Calculated loads at left wing root.

For the F/A-18 configuration modeled, no aircraft maneuver was found for which the nonlinear inertial coupling terms made any difference in traditional responses of interest such as rigid-body motion or wing-root vertical loading. While this study was not exhaustive, the most dramatic differences in predicted results occur precisely where modeling capability is most incomplete. The capabilities to accurately predict aerodynamic loads define the limits of structural model detail that should be attempted.

For nonlinear inertial coupling to become a first-order effect in aircraft, at least one of the following model characteristics should exist: a) aerodynamic loads are low; b) expected rotational rates are of the order of the elastic frequencies; c) model geometry is sufficiently complex that transverse deflection of modeled beams result in

changes in mass distribution. A combined measure for b) and c) is provided by R_j.

No systematic assessment of the effect of the inertial coupling terms on aeroelastic stability was attempted.

VIII. Concluding Remarks

Conventional flight mechanics and aeroelastic analyses have taken advantage of simplifying assumptions regarding the inertial interactions between elastic/angular momentum in model development. These assumptions were based on sound engineering judgement, but occasionally questions have been raised as to the limits of their applicability. In this paper, equations of motion were developed that account for these interactions and generalize the more traditional formulation. The equations use the modes of free vibration as generalized coordinates and define modal inertial loading terms. A candidate parameter, R_j, is defined whose purpose is to assist in determining when elastic/angular inertial coupling should be included in a model. The parameter is a suitably nondimensionalized approximation of quasi-steady deformation due to angular velocity.

The equations of motion developed herein were used to determine open-loop response of a fighter configured without stores. Impact of the inertial coupling terms upon rigid-body response was negligible for this stiff configuration. Both quasi-steady and dynamic effects were observed in certain elastic modes due to rigid-body rotation. The modes affected, however, were aerodynamically uncoupled from the rest of the model dynamics. This decoupling is not guaranteed for all aircraft configurations.

References

[1]Hood, R.V., Dollyhigh, S.M., and Newsom, J.R, "Impact of Flight Systems Integration on Future Aircraft Design", AIAA Paper 84-2459, October 1984.

[2]Schwanz, R.C., "Consistency in Aircraft Structural and Flight Control Analysis", AGARD-CP-228, April 1977.

[3]Waszak, M.R., Schmidt, D.K., "On the Flight Dynamics of Aeroelastic Vehicles", AIAA paper 86-2077, August 1986.

[4]Harschburger, H.E., and Moomaw, R.F., "Experience with the F/A-18 Digital Flight Control System", Presented at the IEEE/AIAA 5th Digital Avionics Systems Conference, Seattle, WA, Oct 31-Nov 3, 1983.

[5]Yurkovich, R., "Flutter of Wings with Leading Edge Control Surfaces", 27th Structures, Structural Dynamics and Materials Conference, Part 2. San Antonio, TX, May 19-21, 1986.

[6]Miller, G., Wykes, J., and Brosnan, M., "Rigid Body-Structural Mode Coupling on a Forward Swept Wing Aircraft", AIAA Paper 82-0683, May 1982.

[7]Ibrahim, R.A., Woodall, T.D., and Heo, H.; "Modal Analysis of Structural Systems involving Nonlinear Coupling"; The Shock and Vibration Bulletin, Part -- 3, Naval Research Laboratory, Washington, D.C., June 1984

[8]Barr, A.D.S., "Some developments in Parametric Stability and Non-linear Vibration", Proceedings of International Conference on Recent Advances in Structural Dynamics, Univiversity of Southampton, England, July 7-11, 1986, pp 545-568.

[9]Cavin, R.K., and Dusto, A.R., "Hamilton's Principle: Finite-Element Methods and Flexible Body Dynamics", AIAA Journal, vol 15, no 12, December 1977, pp 1684-1690.

[10]Dusto, A., Brune, G.W., Dornfield, G.M., Mercer, J.E., Pilet, S.C., Rubbert, P.E., Schwanz, R.C., Smutny, P., Tinoco, E.,N., Weber, J.A., "A method for predicting the stability characteristics of an elastic airplane, Volume 1: FLEXSTAB theoretical description", Boeing Commercial Airplane Co., NASA-CR-114712, October 1974.

[11]Likens, P.W., "Analytical Dynamics and Nonrigid Spacecraft Simulation", Technical Report 32-1593, Jet Propulsion Laboratory, C.I.T., Pasadena, CA, July 15, 1974.

[12]Bisplinghoff, R.L., and Ashley, H., Principles of Aeroelasticity, Dover Publications, 1975.

[13]Whetstone, W. D., "EISI-EAL Engineering Analysis Langauge Reference Manual," Engineering Information Systems, Inc., San Jose, CA, July 1983.

[14]Arbuckle, P.D., Buttrill, C.S., and Zeiler, T.A., "A New Simulation Model Building Process for Use in Dynamics Integration Research", AIAA Paper 87-2498-CP, August 1987.

[15]Peele, E.L., and Adams, W.M.,Jr., "A Digital Program for Calculating the Interaction Between Flexible Structures, Unsteady Aerodynamics, and Active Controls," NASA TM-80040, 1979.

[16]Adams, W.M., Jr., Tiffany, S.H., Newsom, J.R., and Peele, E.L., "STABCAR - A Program for Finding Characteristic Roots of Systems Having Transcendental Stability Matrices," NASA TP-2165, June 1984.

[17]Waszak, M.R, and Schmidt, D.S., "Analysis of Flexible Aircraft Longitudinal Dynamics and Handling Qualities, Volume I - Analysis Methods", NASA CR-177943, June 1985.

[18]Pototzky, A.S., and Perry, B. III, "New and Existing Techniques for Dynamic Loads Analysis of Flexible Airplanes", Journal of Aircraft, Vol.23, No.4, April 1986, pp 340-347.

Table 2. Modal data

	Symmetric modes			Antisymmetric modes		
Mode	M_{jj} slug-ft^2	ω_j rad/sec	$[\Delta J]_j/M_{jj}$ (slug-ft^2)/(slug-ft^2)	M_{jj} slug-ft^2	ω_j rad/sec	$[\Delta J]_j/M_{jj}$ (slug-ft^2)/(slug-ft^2)
1	20.2	26.4	$\begin{vmatrix} -1.76 & .0 & .206 \\ .0 & -1.94 & .0 \\ .206 & .0 & -.198 \end{vmatrix}$	13.2	40.9	$\begin{vmatrix} .0 & 2.00 & .0 \\ 2.00 & .0 & 5.48 \\ .0 & 5.48 & .0 \end{vmatrix}$
2	6.94	53.6	$\begin{vmatrix} 1.04 & .0 & -.041 \\ .0 & 1.04 & .0 \\ -.041 & .0 & .078 \end{vmatrix}$	72.7	51.0	$\begin{vmatrix} .0 & -15.5 & .0 \\ -15.5 & .0 & -.449 \\ .0 & -.449 & .0 \end{vmatrix}$
3	27.7	59.2	$\begin{vmatrix} .773 & .0 & -.799 \\ .0 & .763 & .0 \\ -.799 & .0 & -.007 \end{vmatrix}$	9.61	56.2	$\begin{vmatrix} .0 & 1.51 & .0 \\ 1.51 & .0 & -3.55 \\ .0 & -3.55 & .0 \end{vmatrix}$
4	5.94	74.8	$\begin{vmatrix} 24.0 & .0 & -1.47 \\ .0 & 11.1 & .0 \\ -1.47 & .0 & 35.1 \end{vmatrix}$	5.94	76.4	$\begin{vmatrix} .0 & 26.3 & .0 \\ 26.3 & .0 & .646 \\ .0 & .646 & .0 \end{vmatrix}$
5	27.1	85.1	$\begin{vmatrix} 1.29 & .0 & -.082 \\ .0 & 1.75 & .0 \\ -.082 & .0 & .695 \end{vmatrix}$	1.48	83.2	$\begin{vmatrix} .0 & -1.38 & .0 \\ -1.38 & .0 & -4.00 \\ .0 & -4.00 & .0 \end{vmatrix}$
6	1.35	86.3	$\begin{vmatrix} -2.15 & .0 & -.928 \\ .0 & -2.12 & .0 \\ -.928 & .0 & -.913 \end{vmatrix}$	32.2	95.2	$\begin{vmatrix} .0 & -32.5 & .0 \\ -32.5 & .0 & -.824 \\ .0 & -.824 & .0 \end{vmatrix}$
7	26.1	101.	$\begin{vmatrix} 18.7 & .0 & 1.76 \\ .0 & -13.3 & .0 \\ 1.76 & .0 & 5.51 \end{vmatrix}$	9.40	102.	$\begin{vmatrix} .0 & -11.6 & .0 \\ -11.6 & .0 & 12.2 \\ .0 & 12.2 & .0 \end{vmatrix}$
8	2.63	116.	$\begin{vmatrix} -8.68 & .0 & .056 \\ .0 & 5.63 & .0 \\ .056 & .0 & -16.1 \end{vmatrix}$	7.35	118.	$\begin{vmatrix} .0 & -5.45 & .0 \\ -5.45 & .0 & -1.66 \\ .0 & -1.66 & 0 \end{vmatrix}$
9	27.9	127.	$\begin{vmatrix} -1.07 & .0 & -1.47 \\ .0 & 4.05 & .0 \\ -1.47 & .0 & 3.32 \end{vmatrix}$	21.1	139.	$\begin{vmatrix} .0 & 11.5 & .0 \\ 11.5 & .0 & -.459 \\ .0 & -.459 & .0 \end{vmatrix}$
10	10.8	172.	$\begin{vmatrix} -1.50 & .0 & -.299 \\ .0 & -1.59 & .0 \\ -.299 & .0 & -2.30 \end{vmatrix}$	27.8	146.	$\begin{vmatrix} .0 & -7.40 & 0 \\ -7.40 & .0 & -1.22 \\ .0 & -1.22 & .0 \end{vmatrix}$

SIMULATION IN SUPPORT OF ADVANCED COCKPIT DEVELOPMENT

P.A. Lypaczewski, Eng.
Manager,
Avionics Systems
CAE Electronics Ltd.
Montreal, Canada

A.D. Jones
CSRDF Project Manager
Flight Systems & Simulation
Research Division
NASA Ames Research Center
Moffett Field, CA.

Maj. J.W. Voorhees, Ph.D.
CSRDF Research Manager
Aeroflightdynamics
Directorate, AVSCOM
Moffett Field, CA.

Abstract

The system complexity and high workload of the next generation of light scout/attack (SCAT) helicopters is an area of great interest to the U.S. Army. The Crew Station Research and Development Program has been established by the Army to study the issues of battle captain performance for one-man versus two-man crews. A Crew Station Research and Development Facility (CSRDF) has been contracted for. It consists of a distributed computer system with several stations which play different roles in experiments. Coordination of experiments is done from the Experimenter. Operator Console where a team of Army experimenters and NASA personnel control and monitor the mission scenario used to test the crew members.

Introduction

The system complexity and high workload of the next generation of light scout/attack (SCAT) helicopters is an area of great interest to the U.S. Army. In response to this, the Crew Station Research and Development Program has been established to study the issues of battle captain performance for one-man versus two-man crews when confronted by a hostile environment. The experiments will be conducted at the NASA Ames Research Center by the Crew Station Research and Development Office with the support of NASA's Flight Systems & Simulation Research Division. These groups have contracted CAE Electronics Ltd. of Montreal, Canada to build and integrate the Crew Station Research and Development Facility (CSRDF) for the running of these experiments. Combat Mission Scenario (CMS) software has been developed for this facility by Flight Systems, Inc. of Newport Beach, California to provide the tactical environment for the CSRDF.

Facility Description

The Crew Station Research and Development Facility is a distributed system consisting of several stations which are used to support the Army experiments. The pivotal element of the facility is a two seat tandem helicopter cockpit which provides a realistic environment to the test pilots taking part in the experiments. Three Blue/Red team stations add to the scenario realism by simulating other aircraft on the battlefield while a White station is used to simulate supporting forces with which the crew are expected to interact during the engagement. The coordination of the experiment is done from the Experimenter/Operator console where a team of Army Experimenters and NASA personnel control and monitor the mission scenario that is used to test the crew members.

The design of the various stations in the CSRDF has been undertaken with the requirements of Army experiments in mind. This has been accomplished by taking into account the fidelity of simulation that is necessary as well as the need for ease of use and flexibility to support a research environment.

The tandem station (as shown in Figure 1) has been designed to represent the technology which is expected to be available in the mid 90's. The primary flight instrument for the pilot is a wide field of view Fiber-Optic Helmet-Mounted Display (FOHMD) which presents a panoramic view of the world coupled with sensor outputs and symbology for pilotage, threat alerts and weapon release. Programmable display push buttons allow rapid input to critical aircraft systems such as weapons and countermeasures. Control of the aircraft systems is effected using "glass cockpit" Systems Management Displays (SMDs) via the various tactile data entry devices such as touch pads and touch screens (as shown in Figures 2 and 3). Monitoring of the aircraft situation may be done with the Tactical Situation Display (TSD) that displays a scalable plan view of the gaming area along with several overlays showing the status of threats and friends. These may be modified using the touch screen, as may the navigation and tactics overlays that may also be shown on the TSD. Any of the cockpit controls may also be activated through the interactive voice input system. As well, a comprehensive voice output system keeps the crew members well informed on the status of their aircraft systems.

The flight controls in each crew station consist of two four-axis limited-displacement hand controllers plus foot pedals (allowing full control in each crew position). The longitudinal, lateral, directional and collective controls may be assigned to any combination of the hand controllers and pedals in a given crew position. This reconfigurability allows for the investigation of the impact of various control configurations on crew member efficiency. The flight controls are also dynamically selectable between the front and rear crew positions in order to support the two-man configuration. In order to properly task the crew members in all regimes of flight, the flight control simulation interacts with a real-time blade element rotor model. Given that nap-of-the-earth (NOE) flight will be used for over ninety five percent of the

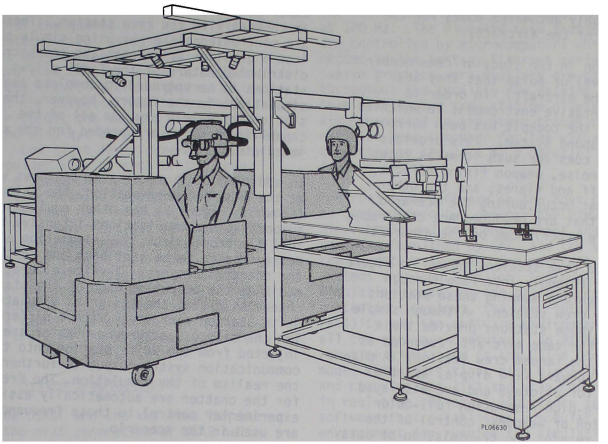


Figure 1. Crew Station Structure

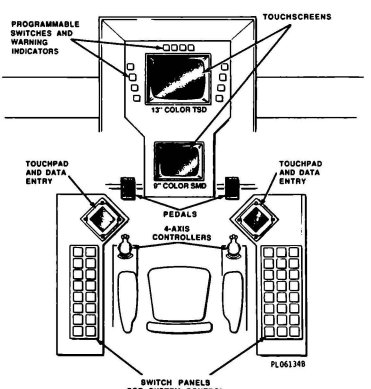


Figure 2. Front Crew Station

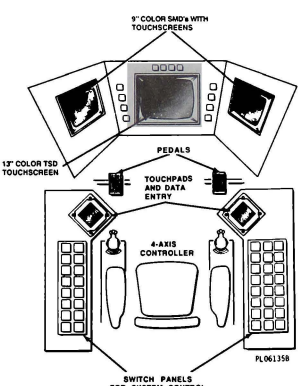


Figure 3. Rear Crew Station

mission, the faithful simulation of the transitioning effects that the rotor model provides is particularly important for the Army experiments. This is coupled with a simulation of an advanced fly by wire flight control system which provides highly automated modes for the effective control of the aircraft.

A key consideration in the study of crew member fatigue is the level of noise that they are subjected to on the aircraft. In order to provide a representative environment in which to do these studies, the cockpit has been surrounded by a six-channel sound system. This provides directional sound cues for such items as rotor and transmission noise, weapon firing effects, dispensing of chaff and flares, and the other various noises that occur during the scenario. The noise levels that are produced are comparable to those which are experienced on the aircraft.

The three Blue/Red team stations (shown in Figure 4) furnish a user friendly interface through which the experimenters manning these stations interact with the crew station. Although simple in construction, these stations provide their pilots precisely the same aircraft resources as are available in the tandem crew station. A plan view or stylized forward view display are the chief references for pilotage and are displayed on a 19-inch ultra-high-resolution full-color monitor. Selection of weapons, control of the flight modes and receipt and transmission of data link messages are all done using soft-key selections on the display's touch screen. Control of the communication system is via a programmable plasma touch panel display. Highly automated flight modes are available for the experimenter's use to allow the stations to fly in concert with the crew station. These modes are controlled either by using simple commands from a joystick or through the touchscreen. The distributed nature of the CSRDF allows these stations to be upgraded to complete team member simulations in the future, however, the simple stations presently provide all of the capabilities which are needed for the system's immediate goals.

In addition to those display features which are at the Blue/Red team station, the White station (shown in Figure 5) has other equipment to support it's communications-intensive task. A full electronic data link simulation allows message pages to be sent back and forth from all the scenario players. A voice alteration unit modifies the output of each of the station's ten channels so that the ten units simulated at the White station each speak with a different voice. Ten channels of background chatter are also injected from the White station onto the communication system output to further augment the realism of the simulation. The frequencies for the chatter are automatically assigned under experimenter control to those frequencies which are used in the scenario.

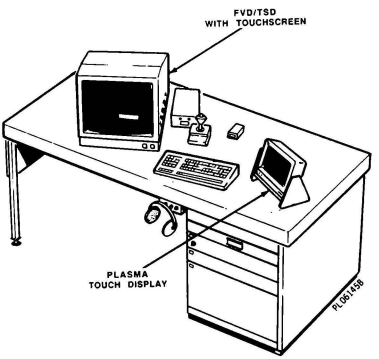


Figure 4. Blue/Red Team Station

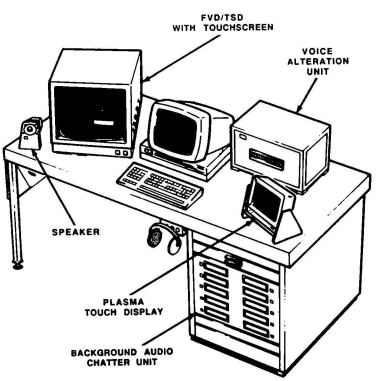


Figure 5. White Team Station

The running of the experiment and the data recording and analysis are conducted from the experimenter/operator console (shown in Figure 6). A myriad of tools are at the experimenter's disposal for monitoring and controlling the experiment. Any part of the gaming area may be viewed under experimenter control using a plan view similar to that used in the crew station, or a stylized forward view display which is used to give a "God's eye view". Ultra-high-resolution monitors repeat the outputs from the computer image generator (CIG), the various crew station displays and the displays at each of the Blue/Red team stations. Low light level cameras display an over-the-shoulder view of each of the crew positions. An eight-channel studio-quality audio recorder saves time-stamped recordings of the mission communications for later analysis. Data recording utilities save critical parameters for analysis after each experiment. Control of all elements of the experiment is made possible by using programmable control pages in conjunction with touchscreens mounted on 19-inch color monitors or via keyboards and plasma touch-panel displays. In this manner the experimenter can effectively keep track of the experiment as it progresses.

System Simulation

The heart of the CSRDF and the key item for the simulation of the next generation of SCAT helicopter is the Fiber-Optic Helmet-Mounted Display (shown in Figures 7 and 8). Through the FOHMD, it is possible to present the pilot with an instantaneous horizontal field of view of 127 degrees and a field of regard that is effectively unlimited. Using an infrared optical head tracker, the pilot's head position is monitored by the host so that the visual scene displays precisely where the pilot is looking. Accelerometers on the pilot's helmet allow lead predictions to be calculated in the host to counteract latencies in the processing and visual image generator loop. This results in a stable, flyable image which tracks all of the pilot's head movements. A high-resolution inset is used to simulate sensor video such as Forward Loading Infra Red (FLIR) from a mast-mounted sight merged with HUD like symbology. A future enhancement to include eyetracking will permit the inset to be used as a high-resolution eye-slaved area-of-interest (AOI) display. The background channels of the FOHMD are used to simulate a wide field of view night vision pilotage system (NVPS) using FLIR imagery generated by the CIG. The visual system data base is modeled to provide accurate pilot cues for NOE flight, which is further reinforced by the texturing of the visual scene. Full-color daytime out-the-window, FLIR and daylight television (DTV) images are available to the pilot from the CIG output channels as viewed from the FOHMD. The visual system interacts with the host computer to show the crew station position within the gaming area and to display weapon effects and target positions in concert with the tactical scenario which runs in the host computer.

A distributed multiprocessor system has been set up to support the CSRDF as shown in Figure 9. This consists of a host computer system which controls and schedules the various satellite processors driving the crew station, team stations and the EOC. The host computer system iterates at a basic rate of 60 Hz coupled with the real-time blade element rotor model running at 120 Hz. The Blue/Red team stations and EOC are controlled by microcomputers which are connected to the host by Ethernet serial data bus. Twelve graphics work-stations are distributed throughout the facility for driving the various team station and crew station displays and are also connected to the other processors by Ethernet. The host computer for the visual system is connected to the host by a parallel OMA data link.

The real-time software in the host computer may be divided into two groups: the simulation of the air vehicle and crew environment and the system software furnished by CAE, and the simulation of the threat environment and aircraft tactical systems supplied by Flight Systems Inc. The CAE simulation comprises the basic aircraft, the nontactical systems, the user interfaces for the facility and the software environment in which all the software executes. An example of the run-time simulation is the blade element rotor model which partitions each blade into segments and computes the lift and drag for each segment in real time. Of equal importance is the system software that establishes a modular, adaptable environment for running the simulation software. This supports the linking and configuration of the simulation in a way that simplifies the task of changing models or modifying existing ones. Using well-defined interfaces into the various models permits the adaptability that is essential for an experimental device.

The responsibility of the mission scenario software developed by Flight Systems is the simulation of the tactical environment. One example is the distribution of threat sites in the gaming area which can simulate a combination of up to one hundred and ten players (including tanks, SAM sites and anti-air-artillery (AAA) sites). The tactics software also simulates the crew and team station weapons and aircraft survivability equipment (ASE), as well as the simple flight models which run in the Blue/Red team stations.

The experimental nature of the CSRDF dictates that the system design could not be "carved in stone" to suit the first set of tests. In order to support future experiments, it is necessary that the facility be easily reconfigurable. To that effect, editors have been furnished which allow all of the user interfaces to be easily modified. An interactive graphics editor allows new crew station displays to be built and existing ones to be changed and linked into the simulation in a simple and rapid fashion. A syntax editor allows the syntax trees which drive the crew station voice input and output systems to be modified to suit a wide range of experiments. Data base processors automatically extract the terrain information from the visual system data base to build the forward view displays, TSD contour maps and tactical line of sight (LOS) data bases used in the facility. Utilities are also being furnished which allow the threat lay down and characteristics to be modified between

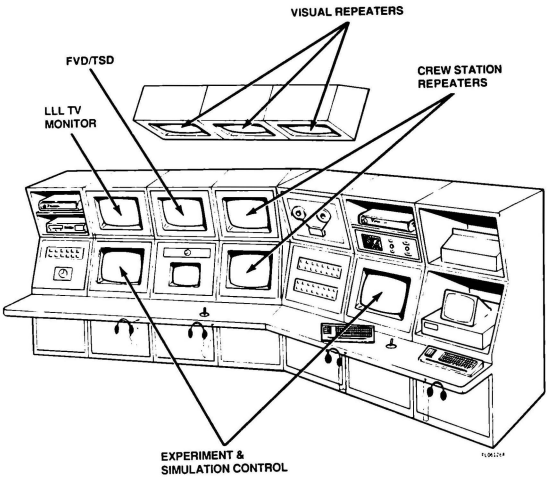


Figure 6. Exerimenter/Operator Console

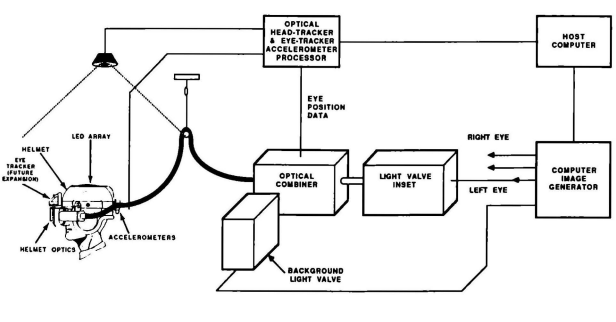


Figure 7. FOHMD System

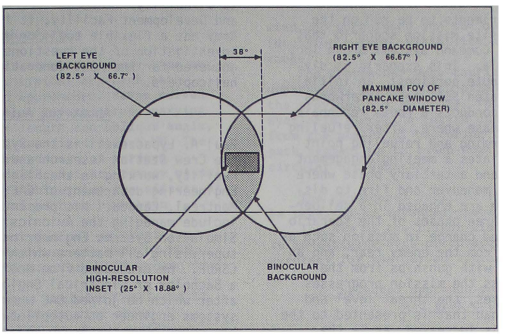


Figure 8. FOHMD Field of View

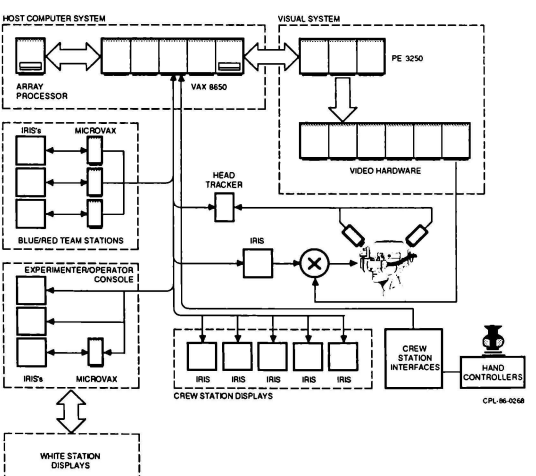


Figure 9. CSRDF Computer Architecture

experiments. Using these sorts of software tools, the facility may be radically reconfigured in a very short period of time.

CSRDF Experiments

The initial Army experiments to be run on the CSRDF utilize a composite mission scenario that has been developed by Command Systems Group Inc. of Torrance, California. This scenario is divided into three 45-minute portions: an initial phase where the simulated SCAT team interworks with air cavalry and conducts a route reconnaissance; a subsequent phase where, after refueling at the forward area arming and refueling point (FARP), the team initiates a meeting engagement with hostile forces; and a tertiary phase where the SCAT team employs maneuver and fire to disrupt enemy units which are engaged in a deliberate attack. In all three phases of the scenario there is an unscheduled change in mission such as extracting personnel from the enemy rear, and an air-to-air engagement with gunships from the enemy forces. Also, as the mission progresses through the three phases, the threat level and communications work load that is presented to the pilot/battle captain is increased, as is the level of fatigue of the crew members. The capability also exists for malfunctions to be inserted to degrade the simulated aircraft systems.

Since the mission scenario is based on the employment of a team of SCAT helicopters, it is essential that those members be realistically simulated in order to interact with the battle captain. This role is fulfilled by the three Blue/Red team stations and the White station. Each Blue/Red team station can be configured to simulate a section of up to four friendly (Blue) aircraft or as enemy (Red) aircraft for the air-to-air engagements. When running in the Blue mode, the experimenters manning those stations interact with the crew station as section leaders and communicate verbally via the simulated aircraft radios or by using the data link simulation.

The White station completes the cast of the scenario participants. The experimenter at this station simulates ten other units which interact with the crew station. These include elements such as the Ranger unit to be rescued, AWACS aircraft that warn of approaching Red aircraft, and the headquarters for ground-based artillery and air cavalry to which the battle captain must transmit and receive reports either verbally or by data link.

The composite mission scenario provides a realistic workload against which the performance of the battle captain can be studied and compared under various circumstances. By configuring the crew station to run with either one or two crew members, comparisons on the effectiveness with which the mission is accomplished can be made. The effect of different cockpit layouts on battle captain performance can also be studied by using either the front or rear positions for that function.

Conclusion

Given the hostile environment that crews are expected to face in the modern battlefield, it is crucial that the aircraft man-machine interfaces be configured in such a way as to optimize survivability. Through the Crew Station Research and Development Facility, it is felt that the Army has a flexible tool for an effective investigation of the questions that must be answered for the next generation of light SCAT helicopters.

About the Authors

Paul A. Lypaczewski is the Systems Architect for the Crew Station Research and Development Facility, working in the Flight Simulation Engineering department of CAE Electronics Ltd., Montreal, Canada. His present responsibilities include managing the Avionics Department of Simulation Systems Engineering at CAE as well as supervising all matters which pertain to the CSRDF. He graduated from McGill University with a Bachelor of Electrical Engineering degree, after which he joined CAE where he worked as a systems engineer on autopilot simulations. He has since been the Group Leader for several tasks, including the integration of advanced airborne equipment onto simulators and visual system development. Mr. Lypaczewski has presented papers on the CSRDF and on the installation of avionics on advanced flight training simulators and has lectured part time at Concordia University.

Maj. James W. Voorhees, Ph.D. is currently serving as the Technical Manager of the Crew Station Research and Development Office of the U.S. Army Aeroflight-dynamics Directorate at Moffett Field, California. Maj. Voorhees is a master Army aviator with two combat aviation tours in the Republic of Vietnam. He received his Ph.D. in experimental psychology in 1980 from Texas Christian University in Fort Worth, Texas. He is the author of over twenty-five papers in the area of Aviation Human Factors, and represents the directorate as simulation and flight test evaluation chief for the Army's Advanced Rotorcraft Technology Integration (ARTI) Program.

A. David Jones is currently assigned to the Flight System and Simulation Research Division of the NASA Ames Research Center, Moffett Field, California., In his current assignment as Project Manager of the Crew Station Research and Development Facility, he is responsible for the design, development, integration and initial operation of this new capability. His most recent assignment was as Chief of the Simulation Investigation Branch. He also served as Facility Manager for the development of the Vertical Motion Simulator and has more than twenty-years experience in the development, operations and management of real-time piloted flight research simulators. Mr. Jones received his Bachelor of Science degree in Aeronautical and Astronautical Engineering in 1964 from the University of Illinois.

A FLIGHT SIMULATOR EVALUATION OF THE APPROACH PATH PARAMETERS FOR MLS CURVED APPROACHES

Louis J.J. Erkelens*
National Aerospace Laboratory NLR
Anthony Fokkerweg 2
1059 CM Amsterdam
The Netherlands

Abstract

The present flight simulator study aims at the evaluation of the typical approach path parameters for curved approaches. Seven curved approach paths have been investigated, varying in both final segment length and oblique angle, while the turn radius was fixed at 1.5 NM. The investigation was carried out for various operating minima (decision height, runway visual range and visibility), while also the effects of wind, turbulence and cloud base have been considered.
The approaches were carried out as precision approaches, during which use was made of MLS guidance, derived from simulated MLS ground facilities.
The simulated conditions included Cat.I and II visibility conditions, while wind shear gradients up to 10 kt/100 ft were considered. Most of the approaches were carried out manually (flight director guidance), although also auto-pilot approaches were included in the test program. Approximately 350 test runs were made. The experimental results consisted of both objective and subjective data. The objective data concerned statistical data of path deviations, aircraft state and control deflections. The subjective data were derived from pilot effort ratings, questionnaire responses and pilot comments.

1 Introduction

The wide coverage volume of the Microwave Landing System (MLS) enables the introduction of new approach and terminal area (TMA) procedures (see fig.1).
For several years the National Aerospace Laboratory NLR has been investigating the potential of MLS with respect to the above-mentioned procedures. During preceding simulation studies at various institutions attention was paid to the feasibility of curved or segmented approach paths and to the possibility of using MLS as a means to provide guidance during the interception phase of these approaches[1-7].

The present flight simulator investigation** has been addressed to the evaluation of the typical parameters for curved approach paths, i.e. length of the straight final segment and magnitude of the turn angle, (see fig.2), in dependence of meteorological conditions such as wind and turbulence and the operating minima (decision height, visibility and runway visual range).

The approaches were carried out as precision approaches, during which primarily use was made of the provided MLS guidance. This guidance was available along the entire curved approach path, from the moment of capturing the oblique approach path segment up to touch down, including the circular arc segment.

The procedures outlined in the paper have to be regarded as advanced procedures for MLS.

The aim of the flight simulator experiment was to investigate both pilot acceptance and tracking accuracy for the curved approach paths.

2 Selected approach path configurations and test conditions

The curved approach path geometry is defined by the following parameters: turn radius R, oblique angle ω_o and length of the straight final segment l_F or final segment altitude h_F. From a previous MLS simulator[7] investigation a turn radius of 1.5 NM was recommended for turns during the final approach phase. Therefore this radius was selected for the present approach paths.

From the multitude of possible approach path geometries a selection had to be made in order to end up with a manageable number of approach paths, which could be investigated within the proposed simulation program.

Finally a series of seven approach paths was selected. A survey of the approach path parameters is presented in table 1. A fixed glide slope of three degrees was applied to all considered approach paths. The magnetic orientation of the simulated runway was 062 degrees.

Regarding the test conditions a selection had to be made of:
- decision heights (DH),
- visibility conditions,
- wind and turbulence models.

As far as the decision heights are considered, the following selections were made:
a) DH at 100 ft, conform the Cat.II requirements
b) DH at 200 ft, conform the Cat.I requirements
c) DH just before the end of the turn to final (DH_{TURN})
d) DH just before the turn to final (DH_{OBL}).

In the cases a) and b) the cloud base values were defined 50 ft above the decision heights. For the cases c) and d), where the decision height is not on straight final, a wider margin of 200 ft was selected, thus giving the pilots some more time to establish the position of the approach and runway lights when coming out of the clouds.

The decision height and cloud base selections have been outlined schematically in figure 3. Concerning the visibility conditions the

---

*Senior research engineer, Stability and Control Department

**The investigation was carried out under contract with the Netherlands Department of Civil Aviation RLD

selections were made as follows: the RVR-values corresponding to the Cat I and II conditions have been applied to the cases a) and b), while for the remaining cases the adjusted visibility was based on the slant range at DH to the runway threshold.

The visibility range was adjusted approximately 500 m in excess of this slant range. Moreover also a fixed visibility range of 5000 m was considered in the test conditions.

Figure 4 shows a survey of the wind profiles as selected for the simulation tests. The different wind profiles contain crosswind situations and various wind shears. Both linear shears with gradient up to 10 kt/100 ft (wind profile 3) and vector shears of 10 degrees/100 ft (wind profile 5) were included. An indication of the severity of wind profile 3 is provided by ICAO[8] which estimates that wind shears in excess of 10 kt/100 ft may be expected on 4 per 1,000 approaches and take-offs. The two vector shears of profile 5 were selected in such a way, that for all approach paths the effect of the turning flight to final was deteriorated by either of these vector shears. The turbulence model, which was structured according to Jansen[9], had been adjusted to a moderate level.

From the multitude of possible combinations of the above-mentioned parameters a limited series of conditions was established. Besides a selection was made of the avionics to be used i.e. autopilot (AP), flight director (FD) and navigation display. A survey of this series of conditions is given in table 2. A distinction was made hereby between conditions to be considered for approach paths having moderate oblique angles (up to 40 deg) and conditions for approach paths with large turn angles. Due to the limited azimuth angle of the visual system of the simulator the conditions with DH on the oblique segment had to be deleted for the approach paths with large turn angles.

3 Simulation facility and mathematical models

Flight simulator equipment

The flight simulation program was carried out on the research flight simulator of the NLR. The simulator equipment consists of several modules:
- a multi-processor computer system,
- a TV-model board visual system,
- a four degrees of freedom motion system,
- a transport type aircraft cockpit, serving a 2-man crew with the possibility of an additional observer seat,
- a control desk,
- recording equipment.

Sophisticated systems and components have been used to achieve a realistic simulation in all considered aspects. Visibility effects such as flying in clouds, haze and fog were introduced by electronically altering the terrain image.

A picture of the cockpit interior showing the instrument panels and controls has been presented in figure 5. The avionics equipment is conventional except for the following items:
- turn annunciator
- along track distance (ATD) indicator
- modified HSI (MLS mode)
- navigation (MAP) display.

Normally an HSI presents the aircraft position relative to the ILS localizer or VOR radial. In the present "MLS mode" the HSI course bar indicates the position of the reference MLS track relative to the aircraft, whereas the course pointer rotates with the momentary direction of the reference track (either straight or circular).

The experimental navigation display had been installed in the centre instrument panel (see fig.6). As shown in figure 6 it presents a plan view of the approach path with numerical presentation of the track angles of the straight path segments, while tick-marks at 2 nautical mile intervals indicate the distance. The bearing of the picture is relative to the aircraft track angle, so it is continuously lining up with the actual track angle. The aircraft reference symbol is fixed in the middle of the lower margin of the screen, the path disappears under the aircraft symbol as the aircraft proceeds along its track.

Besides the pictorial view of the approach path also some peripheral information is presented on the display:
- a track angle scale and pointer
- a digital read-out of the ground speed (kt) and of the true airspeed (kt)
- a vector indicating the wind direction (bearing relative to the aircraft track) and the wind velocity (kt) indicated by 2 digits
- a cross track deviation indicator (dots)
- a glide path deviation indicator (dots)
- a digital ATD read-out (nautical miles).

4 Mathematical models

A simulation model of the Boeing 747 aircraft was used to represent an aircraft type having the dynamic characteristics of a heavy weight type of aircraft. It is expected that this type of aircraft will be critical with respect to flying curved flight paths, due to its large mass and inertia and the higher approach speeds.

For maintaining a desired airspeed the pilots could make use of the available autothrottle system.

The position of the aircraft with respect to the curved approach path was computed by the MLS guidance model using the simulated azimuth, elevation and DME signals.

In order to improve the realism of the simulation, noise was added to the simulated azimuth and elevation signals. For this purpose the noise model as applied in previous simulations was also used in the present simulation. The corresponding noise levels have been based primarily on the ICAO SARP's specification for Annex 10.

The variables used to express the relative aircraft position are (see fig.2):
- along track distance (ATD)
- cross track deviation (CTD)
- vertical track deviation (VTD)

With the use of the ATD, the linear track deviations CTD and VTD are reduced to angular

deviations, which can successively be expressed into the usual "dot" units. For the scaling to "dot" units on the last ~ 6.5 NM of the approach, use has been made of the existing ILS sensitivity scaling specifications. During the intermediate approach phase (distances in excess of ~ 6.5 NM), use was made of a linear deviation scaling. The applied scalings for the lateral and vertical deviation signals are discussed in [7].

The simulated control systems for the autopilot and flight director were capable of capturing and tracking curved paths. The turn algorithm of both systems was based on the closed loop tracking method, which is extensively discussed in [6]. Down to a particular distance from the threshold the drive signals for these control systems are based on angular track deviation signals. Beyond that distance a transition is made to linear track deviations, in order to avoid too high sensitivities of the guidance signals.

5 Description of the test program

The simulator program consisted of sessions of about one hour, during which a crew (pilot and co-pilot) carried out 7 MLS approaches. Seven pilots participated as subject pilots (pilots flying), while eight pilots joined the simulation sessions as co-pilots (pilots not flying). Five of the subject pilots are airline pilots (captains) on wide body transports (Boeing 747, DC-10 and A310), while the other two serve as first officers on Boeing 747 and Boeing 737 aircraft respectively.

Before the actual measurements started the subject pilots were amply able to familiarize with both the simulation equipment and the procedures, before the start of the program.

Due to the large amount of test cases involved and the requirement - for statistical reasons - that each pilot should fly each particular test case 4 times, it was impossible to provide all pilots with the complete matrix of test cases. It was therefore necessary to distribute the different test cases among the subject pilots. Care was taken hereby that each test case was flown by three pilots.

Four replications of each test case (= combination of approach path type and condition) were flown by one pilot. These four identical runs were randomly distributed among a pilot's test program.

Different crew procedures have been applied to approaches with high decision heights (DH above 200 ft) and to approaches with low decision heights (DH of 100 ft and 200 ft).

The flight procedure during an MLS approach, proceeded as follows:
After oblique segment capture this segment is tracked until the 1½ dots below glide slope position is reached.
- At 1½ dots below the glide slope: select gear down.
- At glide slope intercept: select flaps 25°.
- When passing 1500 ft:
  - select landing flaps (30°)
  - reduce to final approach speed (FAS)
  - final checklist.

6 Test results

Besides recorded objective test data, concerning aircraft state and control variables and path deviation variables, also subjective test results were obtained derived from:
- pilot effort ratings
- pilot responses to a questionnaire
- pilot comments.

Objective test results

As an example of the objective data, in figure 7 some of the recorded variables have been plotted as a function of the ATD for a number of test runs. These plots present the data for approach path 01 during 8 flight director approaches. The data were collected for different pilots and different conditions. As appears from these results, the repeatability of the test runs is very satisfactory. The lateral tracking accuracy is very high, even during the turns. The vertical path tracking accuracy (VTD vs ATD plots) appears to be satisfactory too (path deviations within ½ dot). Moreover the interception of the glide path went off smoothly.

In order to present the objective test data in a convenient way, these data have been reduced to statistical results. For the CTD and VTD data, probability contour plots have been determined. The result for approach path 01 is presented in figure 8, for both autopilot and flight director approaches. This figure shows the mean value plots and the contour plots for 95% (2σ) and 99.99999% (6σ) probability respectively, assuming a normal distribution. The superior lateral tracking performance of the autopilot runs is obvious. Roughly speaking, the tracking of the autopilot approaches appears to be a factor 2 better than the flight director approaches.

Another phenomenon to be noticed is that during autopilot approaches no remarkable effect of the turns on the lateral tracking performance is observed. On the other hand, in case of the flight director approaches, a slight deterioration of the lateral tracking is observed during the turns. For both autopilot and flight director tracking performance it could be concluded that the 95% probability contour plots (mean ± 2σ) are all amply within the 1-dot deviation bounds. Likewise the remarks mentioned for approach path 01 apply to the remaining paths.

Another statistical result concerning the tracking accuracy is shown in figure 9. Figure 9a shows histograms of CTD at decision height for both autopilot and flight director tests. Likewise in figure 9b similar results are presented for the VTD. From figure 9a it appears once more that the lateral tracking accuracy in the autopilot runs is obviously better than in the flight director runs, although for both cases the results are highly satisfactory. On the contrary the glide path deviation histogram (fig. 9b) of the autopilot runs indicates that the 1-dot deviation level was exceeded substantially in a large number of cases. Also in case of the flight director histogram the 1-dot level was exceeded frequently. As appeared from a detailed investigation, this was mainly due to the inability of the autothrottle system to manage the strong wind shear (10 kt/100 ft) of windprofile 3. This wind shear also caused the major number of missed approaches (10 out of a total number of 17 missed approaches).

A series of histograms have been determined for the maximum bank angle during the turn to final. According to the histograms in figure 10, the maximum bank angles appearing during the autopilot runs were in between 11° and 17°. In case of the flight director approaches the variation in maximum bank angle was considerably higher (between 9° and 28°). As appears from the distribution, however, in less than 1% of the cases the 25 degrees bank angle was exceeded.

Subjective test results

After completion of each test run the pilots were requested to indicate the experienced workload on an effort rating card. The ratings had to be given according to the McDonnell[10] rating scale for demands on pilot. An analysis of variance was applied to the rating data. One of the results of this analysis is presented in figure 11. The separate contributions of the different experimental parameters to the overall effort rating are shown herein. The major factor appears to be the subject pilot himself. Evidently the effort ratings between the individual pilots A through G differed substantially.

Concerning the ratings for the approach paths, it seems that for the autopilot approaches a slight difference exists between the paths 01 ÷ 04 and the paths 05, 08 and 09. This difference is, however, due to the different wind profiles which were applied. This is illustrated in the lower left diagram ("wind profiles") of figure 11.

In case of the flight director data neither the approach path type nor the wind profile appear to have significant effects on the ratings.

Other subjective data was obtained from the pilot questionnaire. The results have been considered separately for autopilot and flight director results. It appeared that the pilots could evaluate the various questions better for the flight director than for the autopilot runs. This may be attributed to the fact that pilots are more aware of what is going on when they fly in the flight director mode, while it is difficult for a pilot to qualify an automatically flown approach since he has to draw his conclusions merely from monitoring experience.

One question concerned the pilot opinion about the altitude at the beginning of the final segment (h_F). In figure 12a the corresponding results have been plotted versus h_F. The diagram shows that h_F could be as low as 300 ft - 400 ft as far as pilot opinion is concerned.

By means of two other questions, the pilots were able to evaluate the stabilization altitude on final. The relation between the estimated stabilization altitudes (h_{est}) and h_F appears very clearly from the diagram in figure 12b. Besides in this diagram the recommended minimum for the stabilization altitude (h_{rec}) has been plotted.

It appears that stabilization was achieved slightly beyond h_F. From the responses to the question concerning the lowest stabilization altitude it became clear that the pilots were very consistent in their opinion about the magnitude of this altitude. The h_{rec} appears to be slightly more than 300 ft for all the approach paths.

A survey of the pilot opinion about the bank angle, experienced during the turns, is depicted in the bar diagram of figure 13. The bank angle φ is directly related to the turn radius R, according to: $\varphi = \tan^{-1} \frac{V_G^2}{gR}$ (V_G = ground speed). In this way the turn radius is evaluated implicitely by this question. From the response scores shown in the diagram it is concluded that the pilots did not have objections to the magnitude of the bank angles. Consequently the applied turn radius can be considered acceptable. It is noticeable, however, that for approach paths having short final lengths 10% of the answers read "occasionally too steep", whereas the same score was obtained for the answer "occasionally too shallow" in case of the paths with relatively long final lengths. Possibly, the acceptance of bank angle magnitude slightly depends on the altitude at which the turns are flown.

The pilot questionnaire also included an overall evaluation of the curved approach procedures. Within this scope an important result was obtained from the responses to the question concerning the acceptability of the approaches as future standard procedures. In the diagram of figure 14a the responses to this question are shown. An increasing trend in acceptability with increasing final intercept altitude is observed (dashed line). It has to be remarked that even in case of 200 ft and 300 ft final intercept altitude, the answer "acceptable" scored still 40% and 65% respectively.

The question about typical path parameters i.e. final segment length l_f (or altitude h_F) and turn angle ω_p, enabled the pilots to evaluate the approach path geometry. As appears from the bar diagram in figure 14b the final segment altitude h_F was the only important factor. Relevant objections were raised to the final segment altitude h_F, only if this altitude became less than 400 ft (l_F < 1.25 NM).

Moreover a similar trend of decreasing objections with increasing h_F is observed in figure 14b. In figure 15 the questionnaire responses have been plotted, regarding the question of recommended minimum decision height. The result indicates that for approach paths with final intercept altitudes of approx. 400 ft or above, the adjusted decision heights for Cat.I and Cat.II approaches are generally accepted by the pilots as recommended minimum values. For approach paths with lower final intercept altitudes a higher decision height was requested (increments up to 100 ft).

7 Conclusions and recommendations

Summarizing the various experimental results, obtained during the flight simulator investigation, from tracking data, aircraft state and control data, effort ratings, questionnaire responses and pilot comments, the following conclusions can be made:

- The lateral tracking performance of the curved approach paths is very high, even in case of the flight director approaches.

- The vertical tracking performance is also satisfactory, provided that no strong wind shears are encountered.

- The approaches are feasible under both Cat.I and II conditions. For Cat.II an adequate autopilot/autothrottle system is indispensable.

- For MLS-guided curved approaches the decision height should be located on the straight final segment.

- As far as the approach path acceptance by the pilots is concerned, it appeared that:
  . the minimum acceptable straight final segment length is approximately 1.25 NM (final segment altitude 400 ft).
  . the turn angle is restricted by MLS coverage limitations, not by pilot acceptance.
  . the selected turn radius of 1.5 NM kept the bank angle range during the turns within acceptable limits.

- The turns to final have to be flown closed loop, which means that the guidance algorithms for autopilot and flight director steering have to be related to the lateral deviation from the circular reference track, while an additional bank angle command has to be included, in order to compensate for the curvature of the path.

- Wind shear is obviously the single most limiting factor regarding the feasibility of curved approaches.

- A map display format similar to the one simulated is highly appreciated and is preferred by far to the usual HSI format, since it contributes better to the situational awareness.

- It was emphasized by all pilots that conventional autothrottle systems, which operate independently from autopilot and flight director, are less suitable for the execution of curved approach procedures. For these procedures, especially in those cases where the final segment is short and wind shear is encountered, an integrated "wind shear capable" autothrottle, autopilot and flight director system is recommended. For future control systems it is therefore encouraged to consider a new, integrated approach of the autopilot, flight director and autothrottle design.

8 References

1. Sager, D., Simulator evaluation of manually flown curved instrument approaches. Massachusetts Institute of Technology, January 1973, FTL report R-73-1

2. Erkelens, L.J.J., Flight simulation studies on the feasibility of laterally segmented approaches in an MLS environment. NLR MP 82025 U, Amsterdam, 1982. (Paper presented at the 13th ICAS Congress, AIAA Aircraft Systems and Technology Conference, Seattle, USA, August 1982).

3. Branstetter, J.R., Houck, J.A., Use of flight simulation to develop terminal instrument procedures for transport category aircraft. AIAA Atmospheric Flight Mechanics Conference, August 18-20, 1986, Williamsburg, Virginia (Paper no. 86-2072)

4. Knox, Ch.E., Flying complex approach paths using the microwave landing system. SAE Technical Paper 861771, October 1986.

5. Lambregts, A.A., Development of an MLS lateral autoland system with automatic path definition. AIAA Guidance and Control Conference August 19-21, 1981, Albuquerque, New Mexico (Paper no. 81-1751)

6. Erkelens, L.J.J., Geest, P.J. van der, Hagenberg, T.H.M., Schrier, D., Pouwels, R., Investigation on MLS approach path interception and transition techniques. Part I: Fast-time computer simulations. NLR TR 85097 U, Amsterdam, 1985, ICAO-COM/OPS 85, IP/23.

7. Erkelens, L.J.J., Geest, P.J. van der, Investigation on MLS approach path interception and transition techniques. Part II: Flight simulator investigation. NLR TR 85097 U, Amsterdam, 1985.

8. Lord, G.A.B., Wind shear: that sinking feeling. Flight International, December 31, 1977.

9. Jansen, C.J., Non-Gaussian atmospheric turbulence model for flight simulator research. Journal of Aircraft. Vol. 19, nr. 5 May 1982 (AIAA paper 80-1568).

10. McDonnell, J.D., Pilot rating techniques for the estimation and evaluation of handling qualities. Systems Technology, Inc. Hawthorne, California, December 1968.

TABLE 1:
CONSIDERED PATH CONFIGURATIONS

PATH NR	ℓ_F	h_F	ω_0	
	NM	ft	deg	
01	0.63	200	20	approach altitude 2000'
02	0.94	300	20	
03	1.26	400	30	
04	1.26	400	40	
05	1.5	480	60 *	
08	2.0	640	90 *	approach altitude 1500'
09	2.5	800	90 *	

GLIDE SLOPE: 3°

TURN RADIUS: 1.5 NM

*) extended azimuth coverage required (60°)

TABLE 2:
CONSIDERED CONDITIONS PER APPROACH PATH

$\omega_0 \leq 40°$

COND		DH	WIND PROFILE	AP/FD	NAV. DISPL.
①	CAT II	100	1	AP	+
②	CAT I	200	2	FD	+
③		DH_{TURN}	3	FD	+
④		DH_{OBL}	4	FD	+
⑤		DH_{OBL}	5	FD	–

$\omega_0 > 40°$

COND		DH	WIND PROFILE	AP/FD	NAV. DISPL.
①	CAT II	100	3	AP	+
②	CAT I	200	2	FD	+
③		DH_{TURN}	5	FD	–

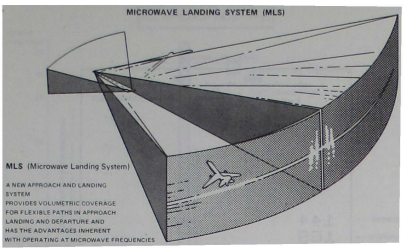


Fig. 1 Signal coverage volume of the Microwave Landing System MLS

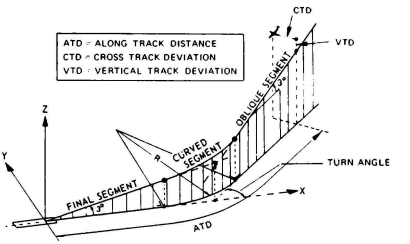


Fig. 2 Lay-out of the curved approach path geometry

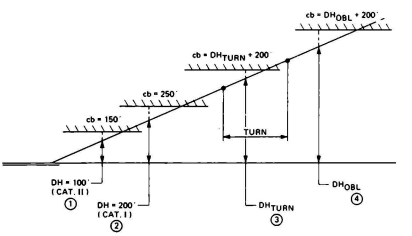


Fig. 3 Selected decision height (DH) and cloud base (cb) conditions

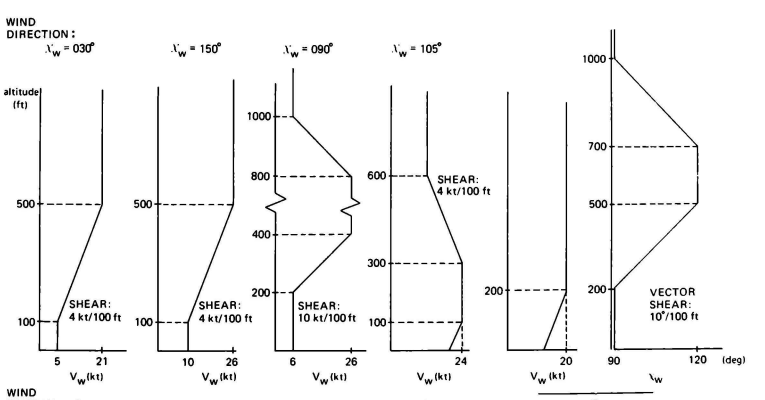


Fig. 4 Simulated wind profiles

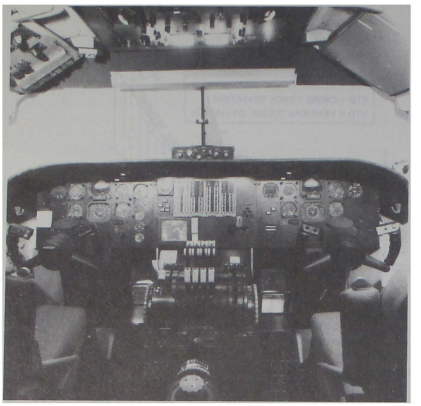


Fig. 5 Picture of the simulator cockpit interior

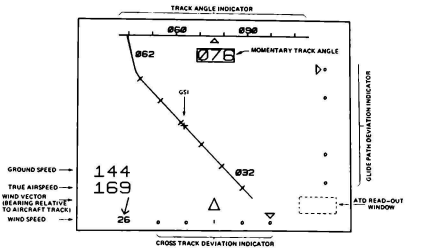


Fig. 6 Example of the picture presented on the navigation display

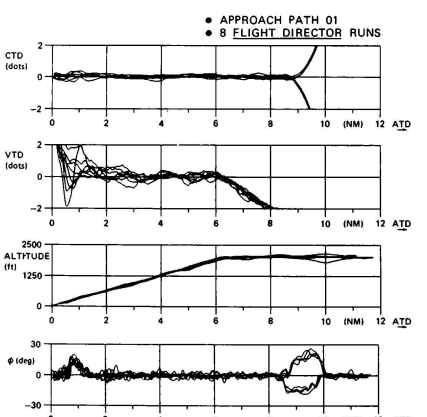


Fig. 7 Plots showing the recorded CTD, VTD, altitude and bank angle as a function of ATD for approach path 01 (flight director runs)

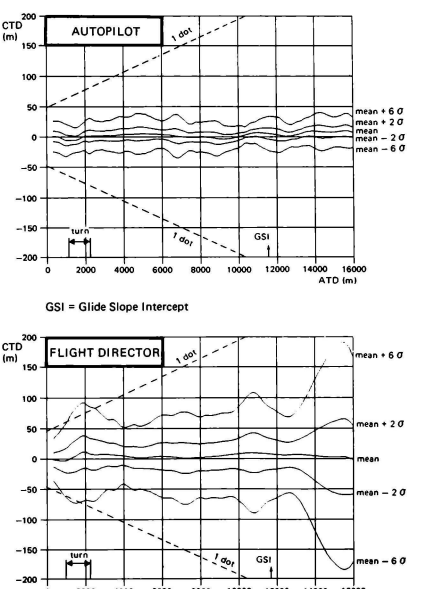


Fig. 8 Lateral path tracking statistics for approach path 01

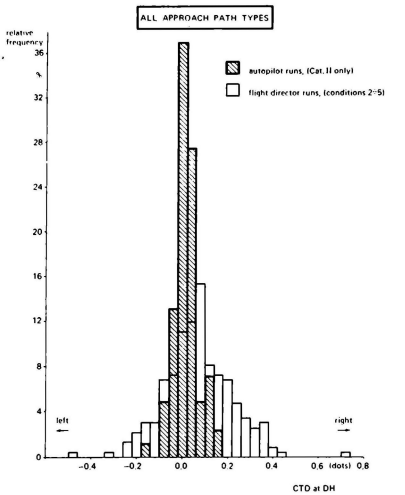


Fig. 9a Histograms of cross track deviation (CTD) at decision height (DH)

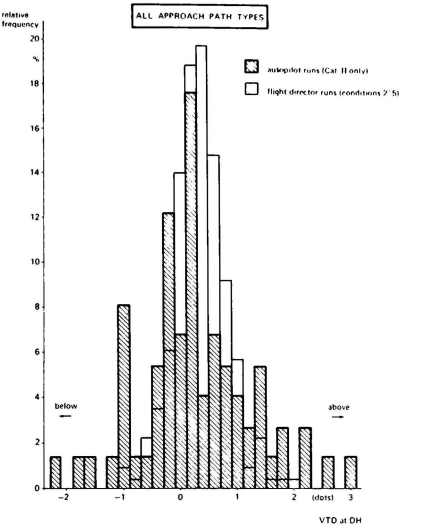


Fig. 9b Histograms of vertical track deviation (VTD) at decision height (DH)

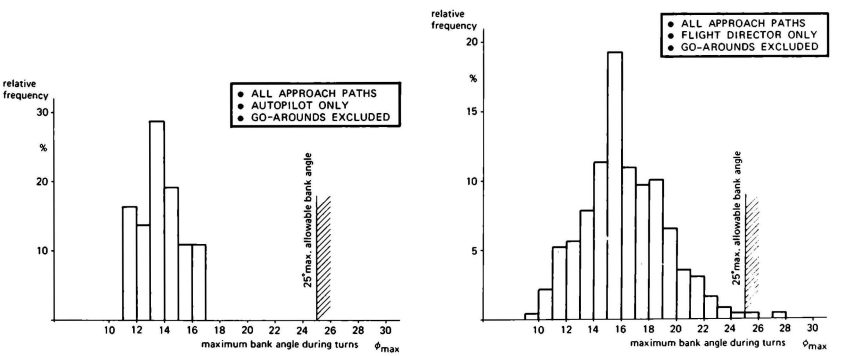


Fig. 10 Histograms of maximum bank angle during turns

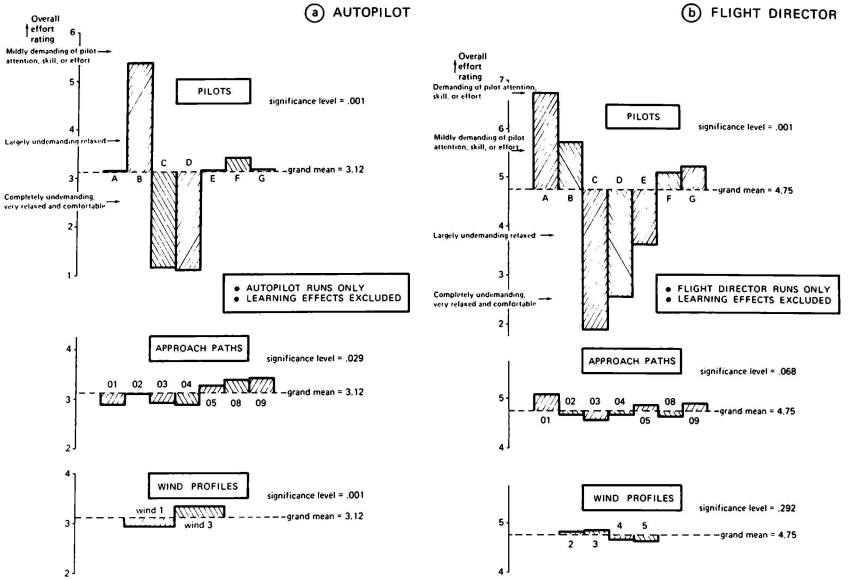


Fig. 11 Results of analysis of variance applied to the overall effort rating

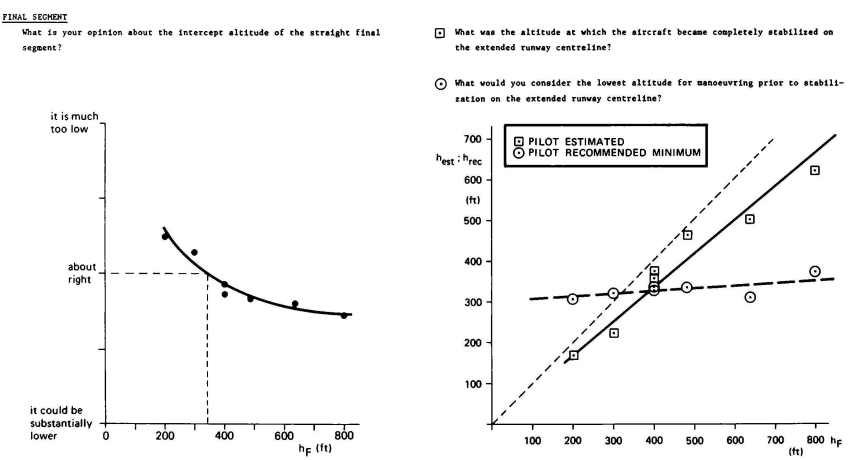


Fig. 12a Evaluation of final intercept altitude

Fig. 12b Comparison of estimated stabilization altitude (H_{est}) with recommended stabilization altitude (H_{rec})

Fig. 12 Questionnaire results (final segment)

FLIGHT DIRECTOR APPROACHES

TURN SEGMENT

How do you evaluate the magnitude of the bank angle during the turn to final?

- too shallow
- occasionally too shallow
- about right
- occasionally too steep
- too steep

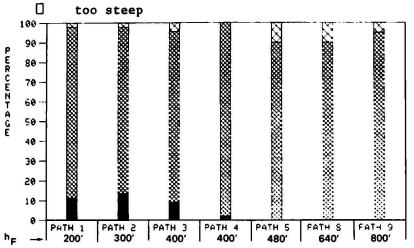


Fig. 13 Evaluation of bank angle during turns

FLIGHT DIRECTOR APPROACHES

TOTAL APPROACH

Do you consider this approach, with corresponding operating minima, acceptable as a future MLS standard procedure?

- yes
- marginal
- no, because .....

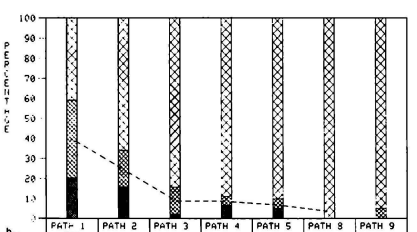


Fig. 14a Acceptability of the curved approach paths

FLIGHT DIRECTOR APPROACHES

What is your opinion about the combination of oblique angle/final intercept altitude?

- No objections to this combination
- Intercept angle too large; altitude correct
- Intercept altitude too low; angle correct
- Combination of both angle and altitude yields problems

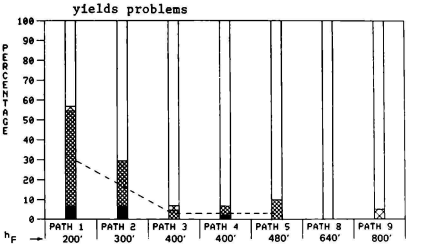


Fig. 14b Evaluation of the curved approach path geometry

Fig. 14 Questionnaire results (total approach)

What operating minima would you recommend for this approach?

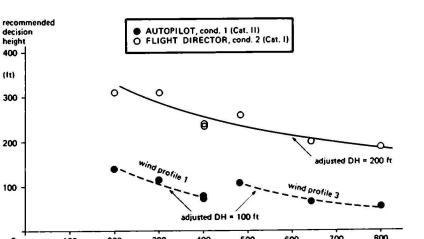


Fig. 15 Recommended decision height (DH_{rec}) as a function of final intercept alitude (h_F)

87-2637

A PROGRAMMABLE GAMING AREA

Kenneth R. Henke*
Research and Development Engineer
Lockheed California Company
Burbank, California

ABSTRACT

Making changes in computer image generating system (CIG) databases is a time consuming laborious job involving changing many mathematical inputs for vertices, colors, shapes, etc. A new menu driven software program has allowed the easy modification of a CIG gaming area with none of the former labor intensive activity.

BACKGROUND

As part of a planned effort to provide a corporate Mission Simulation capability as well as to prepare for the ATF competition, Lockheed set out to design and build a Weapons System Simulation Center in 1983. The result of that effort was a new and unique facility that came on line at the Kelly Johnson Research & Development Center in June of this year.

The WSSC, as the new facility is called, was intended to perform all types of manned simulation required for aircraft design and development. This included cockpit development, handling qualities and, last but not least, mission simulation.

The requirement for piloted mission simulation carried with it the need for visual simulation. Earlier Lockheed experience with model board, TV monitor type of visuals revealed limitations in usage for large scale mission simulation. Similarly, the early versions of computer generated imagery that were available in the early 1980's did not appear to be adequate for the desired capability.

A study was made of the newer generation of CIG's being developed at that time. The General Electric Compu-Scene IV was selected after a detailed evaluation effort as one of the best computer image generators available.

UNDESIRABLE DATABASE FEATURES

The new CIG now needed a new database or gaming area to go with it. Figure 1 shows a list of problems that were associated with the various gaming areas that had been used in the past.

- ONLY SMALL AREAS AVAILABLE
- PRIMARILY AIRPORT AREAS
- DIFFICULT TO MODIFY CULTURAL FEATURES FOR DIFFERENT TESTS
- EXPENSIVE TO MODIFY

Figure 1. Limitations

Each of these problems resulted in an undesirable feature. The gaming area was either too expensive to build or unsuitable for development type simulation. A team of Lockheed "users" came up with a list of criteria that they wanted to see in the Mission Simulator gaming area or areas. The requirements shown in Figure 2 were presented to the design team.

- LARGE GAMING AREAS FOR TACTICAL SIMULATION
- MULTIPLE GAMING AREAS AVAILABLE FOR DIFFERENT MISSIONS
- EASILY RELOCATABLE CULTURAL FEATURES
- MANY DIFFERENT TOPOGRAPHIC FEATURES AVAILABLE

Figure 2. Gaming Area Criteria

NEW TECHNICAL APPROACH

The result was a building block approach to building up a gaming area. Thirty different 40 nautical mile x 40 nautical mile blocks were designed with corners of specific elevations. Standardized elevations of Sea Level, 2000 feet, 6000 feet and 12,000 feet were selected. Since the various blocks had to mate with other blocks having the same corner elevations, the elevation contour had to be standardized for blocks with similar corner elevations.

Topographic contours were laid out to provide realistic elevation lines. It was found that truly realistic contours were not possible in all cases but the necessary compromises were considered acceptable.

GAMING AREA GENERATION

The key to the programmable gaming area is the ability to join the various building blocks including mirror images through the use of menu driven Database Generation System software.

Thus, a mission planner can generate a large gaming area without the necessity to perform the laborious task of vertex identification and color selection as has been the case in the past. Figure 3 shown on the last page of this paper shows how the planner can visualize the gaming area without the need for use of the CIG and/or the DBGS. In this case, the gaming area was put together as demonstration area which used all of the various building blocks plus several mirror image blocks.

DELETABLE CULTURAL FEATURES

One of the desired features was to provide easily modifiable features on the gaming area so that the tactical situation could be changed at will. Figure 4 shows a typical block with rivers, roads and railroads superimposed. Individual sections of these features could be selected by the mission planner to come up with his desired situation. On "de-select", the surface of the block reverted to the same ground cover as the surrounding area. Thus, deletion of a river resulted in a green valley with no river, or deletion of road segment in the desert resulted in a normal desert sagebrush and rocks situation.

This "selection/deselection" is done during the initial gaming area buildup using the menu driven DBGS software program. It appears that

*Kenneth R. Henke is a member of AIAA.

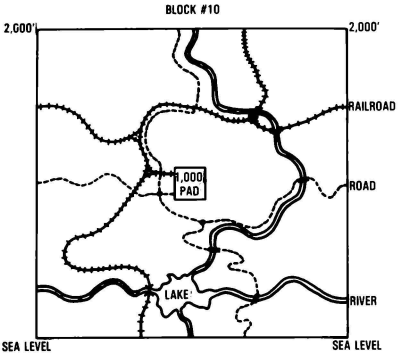


Figure 4. Cultural Features

it would be desirable to expand the technique to allow "selection/deselection" as part of a gaming data base initialization. This could be done prior to a specific simulation run, thus providing a capability to change the basic scene at will during the simulation program.

CULTURAL FEATURE PLACEMENT

Another basic desire was to allow modification or relocation of targets or other significant cultural landmarks. Each block was designed with several flat area "pads". These pads appeared to match with the surrounding countryside, whether it be desert, farmland, or whatever. Model complexes were designed that could be "dropped" into place on a pad, thus resulting in placement of such potential targets as an airfield, factory complex, oil refinery or any one of a number of other selected features as shown in Figure 5. The intent was to allow placement of a particular target, e.g. an oil refinery, in a canyon on one day and on a hilltop or flat plain on another day.

- CITIES, TOWNS
- OIL REFINERIES, OIL FIELDS
- POWER PLANTS
- AIRPORTS
- FACTORY SITES
- RADAR SITES
- ANTI-AIRCRAFT EMPLACEMENTS

Figure 5. Model Complexes

Again, this population of the cultural area pads is done with the menu driven software where the particular model complex is selected, its angular position selected and the placement with roads and railroads incorporated.

Since any tactical ground targets would be expected to be protected by anti-aircraft defenses, each of the model complex "pads" was surrounded by a number of smaller protective "pads". These were placed in various strategic locations. Anti-aircraft installations, radar stations, surface to air missile launchers, etc., were modeled to be placed on these surrounding pads as desired by the simulation scenario planner. Again, the "pad" reverted to the surrounding terrain and ground cover when not occupied The net result here was that any tactical strategy developed during a simulation run or runs could be upset by relocation of the defenses, thus forcing the mission planner or pilot to consider all possible defensive situations in his tactical plans. Figure 6 shows a schematic of one of the building blocks with the various pads shown.

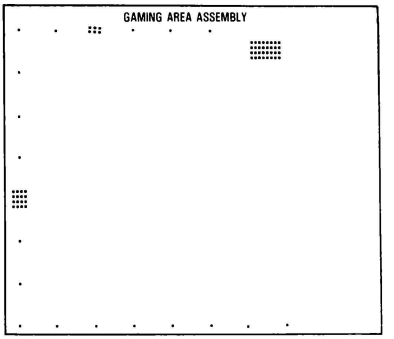


Figure 6. Building Block Pads

CONCLUSIONS

While the concept of the Programmable Gaming Area is new and has not been utilized for a significant period of time, it shows great promise in providing variations in simulation gaming areas without the time consuming and expensive terrain modeling for each variation.

ACKNOWLEDGMENTS

I wish to acknowledge the support of Mr. Ray Goudey, retired Lockheed Test Pilot and Consultant on WSSC and Mr. Hank McConnell of General Electric Simulation Systems of Daytona Beach. Hank was instrumental in transforming some of the basic thoughts and ideas into the actual data base.

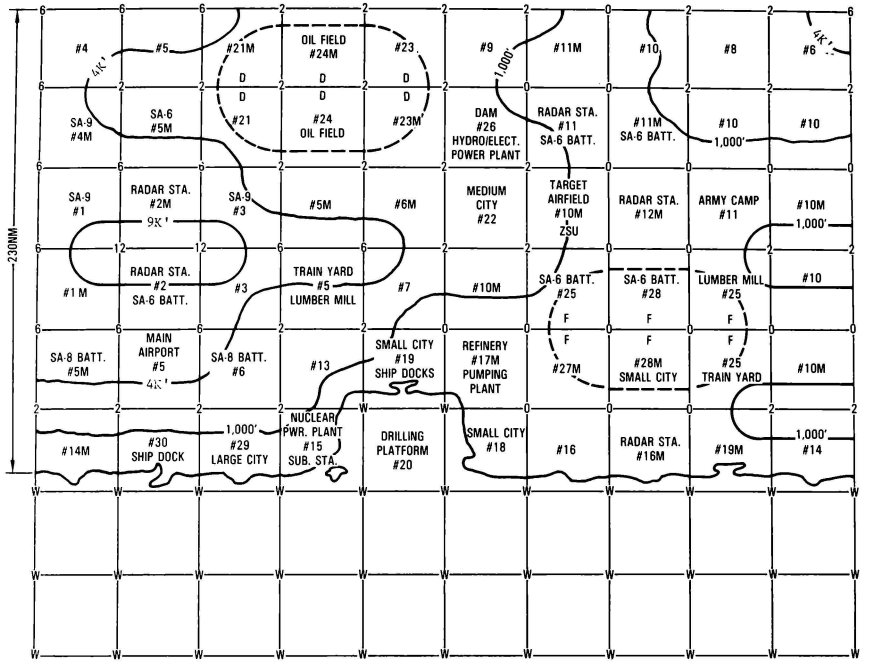


Figure 3. Gaming Area Layout

AH-64 COMBAT MISSION SIMULATOR TACTICAL SYSTEM

Edward Drew, Staff Scientist
Gary George, Staff Engineer
Samuel Knight, Senior Staff Engineer

The Singer Company—Link Flight Simulation Division
Binghamton, New York

Abstract

The AH-64 Apache is the first Army attack helicopter developed to fight autonomously in the forward battle area. It is designed to perform in daylight or at night and in adverse weather, with an emphasis on the capability to fight, survive, and fight again. This fight-and-survive philosophy required unique designs and high-fidelity simulation in Singer-Link's Daedalian Award-winning AH-64 Combat Mission Simulator (CMS).

This paper describes the design of the heart of the CMS, the tactical system. Included are the Fire Control System, Target Acquisition and Designation System (TADS), Pilot Night Vision Sensor (PNVS), Laser Rangefinder/ Designator, Laser Tracker, Integrated Helmet and Display Sighting System, Hellfire Point Target Weapon System, 2.75-inch Aerial Rocket Control and Delivery System, 30-mm gun, and airborne survivability equipment, as well as the interactive threat environment simulated. It also discusses the implementation of these systems in the total mission scenario required by the Army for training.

AH-64 CMS Overview

The AH-64 CMS has set the standard for a new, unique, and extremely effective class of Army training device, the Combat Mission Simulator. Surpassing flight simulators and weapon system trainers, the CMS is a state-of-the-art achievement in developing and integrating high-fidelity simulations of the AH-64 aircraft systems with a large-scale hostile threat environment. The high-fidelity simulations allow the student to be trained to be fully knowledgeable of the tactical capabilities of the AH-64, while the nonforgiving threat environment forces him to fully exploit those capabilities. The instructor can change threat scenarios and move targets into the battle area as would be typical of actual combat. This provides for surprise engagements as well as the requirement for the student to quickly identify targets and mask in preparation for optimal engagement. Determination of optional engagement requires the student to use maps, consider best areas of backdrop, evaluate the total threat scenario as he knows it from both his observations and his scouts, and make the most effective use of his weapons.

The AH-64 CMS employs separate pilot and copilot crew stations, thus allowing both integrated and independent crew training. Each crew station consists of a fully instrumented cockpit with sensor displays; a three-window/ wide-angle, collimated out-the-window display system; an aural cue system; and an on-board instructor/operator station. Each station is mounted on a six-degree-of-freedom motion platform. The crew station simulations are supported by a shared mainframe computer complex and two visual computer complexes. One visual complex provides out-the-window or night vision video, depending on the training mode selected. The other visual complex is dedicated to the generation of sensor imagery.

The aircraft simulation is accomplished by four primary systems: air vehicle, navigation and communications, tactics, and the instructional system. These systems in turn utilize the resources of three additional systems: aural cue, kinesthetic cue, and the visual system. A real-time control system provides overall control through the main computational system.

The air vehicle system provides simulation of the powerplant, aircraft accessory systems (hydraulic, electrical, etc.), and flight control systems, including the Digital Automatic Stabilization Equipment and stabilator, as well as the aerodynamic models for flight dynamics.

Simulated navigational systems include Automatic Direction Finding, the Lightweight Doppler Navigation Set, the radar altimeter, the Heading and Attitude Reference System, and simulation of navigation facilities. Communication simulation includes AH-64 intercommunications and transceiver sets, secure voice, an Identification Friend or Foe system, and ground communication facilities.

The instructional system provides an extensive complement of controls and graphic displays designed to aid the instructor in preparing, monitoring, altering, and reviewing the training scenarios.

The aural cue system provides the crew with air vehicle and tactical sounds. The tactical sounds include gunfire, rocket and missile launch, and near misses and hits on the ownship.

The kinesthetic cue system is utilized by both air vehicle and tactics. This system provides cueing through perturbations to the motion systems and crewmember seat shakers. Tactics kinesthetic cues are implemented in association with gunfire, rocket and missile launch, and near misses and hits on ownship.

The visual system provides out-the-window and sensor imagery of the tactically formatted database, air and ground targets, ownship and threat weapons launch, in-flight, miss and hit effects, and special effects such as tank tracks, dust, IR heat trails, and smoke. Of particular significance to the tactical training is the high-fidelity sensor imagery for forward-looking infrared, day television, and direct-view optics. Separate IR databases and internally designed special effects hardware provide an unprecedented state-of-the-art infrared/television sensor simulation incorporating realistic IR signatures, time-of-day temperature effects, resolution effects, visibility effects, automatic gain control, and AC coupling effects. These parameters are all significant in the training of AH-64 sensor utilization, especially in relationship to locating, identifying, tracking, and engaging threat vehicles.

Of even more importance to the tactical simulation and to the realization of true CMS capabilities is the feature correlation capability of the visual system. Feature correlation refers to the ability to effectively "look into" the database in real time to acquire impact and range information. Feature correlation is utilized in the AH-64 CMS to determine laser rangefinder range returns, laser designation spot position, laser tracker line of sight, target line of sight occulting, ownship mask heights and backdrop, radar warning receiver partial masking, weapons impact positions, missile seeker line of sight, radar altitude, gear height, ownship crash, and rotor crash.

Feature correlation tests are performed by the visual priority and sectoring processor and are commonly referred to as correlation requests. This processor facilitates the unique total-mission capabilities of the CMS by virtue of the large volume of requests it can process: 120 requests per second in the integrated mode using the resources of both digital image generators. Such a high rate is essential to simultaneously support ownship flight, ownship sensors, ownship weapons, and a multiple-target, semi-intelligent threat environment as required to simulate an effective combat mission environment. A resource management system is included in the tactical simulation software to control and optimize the utilization of correlation requests based on such factors as the simulator mode and the number of active targets in the scenario.

The tactical system is organized into five subsystems: fire control, visionics, armament, survivability, and tactical environment. Fire control systems includes stimulation of the Government-furnished equipment fire control computer and data entry keyboard, and simulation of the 1553 multiplexed bus system and remote terminals. Visionics systems include simulation of the target acquisition and designation sight (including stimulation of an actual optical relay tube and actual image autotracker board), pilot night vision sensor, integrated helmet and display sighting system, stimulation of an actual aircraft video display unit and AH-64 symbol generators, and control of simulator-unique video switching networks. Armament systems include simulations for the Remote Hellfire Electronics system, rocket electronics, gun turret and electronics, stores configuration and jettison, fire control panels, and ballistics for each weapon type. Survivability systems include simulation of the APR-39 radar warning receiver, ALQ-136 radar jammer, ALQ-144 infrared jammer, and the M-130 chaff dispenser. Tactical environment systems include target control (including all weapons effects), priority and sectoring processor resource management, ownship scoring, and threat scoring (including the semi-intelligent threat algorithm).

Fire Control Systems

Two aircraft fire control computers are used. Only one computer is enabled in the integrated mode, whereas both are used in independent mode. Unmodified aircraft operational software is resident in these computers. Initial engineering development of the CMS was based on the so-called "blue square" version of software which was being evaluated on prototype Apaches. During development, the CMS converted to the -25 version of software. This conversion provided aircraft concurrency and at the same time demonstrated that the CMS could accept updated operational fire control programs with minimal additional engineering efforts. Incorporation of further updates is anticipated.

The bus system interface from each fire control computer is tied to an aircraft symbol generator as well as a simulator-unique bus interface device. This bus interface unit simulates all the multiplex remote terminal units other than the symbol generator. This type of system allows for bus control for some simulator-unique requirements such as freeze where the symbol generator messages must be maintained.

Another significant design effort was involved in developing a system to allow proper operation during periods when simulator-unique functions are enabled, including freeze, condition store, record, replay, and slow time. Utilizing the

fire control computer's test interface, a dedicated microprocessor and specially designed interface electronics provide access directly to key internal control signals and buses. When a simulator function is selected, the computer is diverted to operating on instructions from an external memory source contained on the microprocessor.

Synchronization of subsystems was a critical design consideration in development of the CMS because of the number of subsystems involved in certain control loops, especially those relating to the visionics systems. Synchronization of the CMS, both visual and mainframe systems, is controlled by the FCC.

The fire control system requires entry of mission-related variables such as manual range, laser codes, target coordinates, and fault detection commands. These are done using an actual data entry keyboard with special interface hardware. The simulation also includes software to provide the fire control computer with instructor-inserted target waypoints, magnetic variation, and present position either as an initial condition or as edited during the mission. The software performs this function by emulating data entry to the fire control computer.

Visionics System

The visionics system consists of the Integrated Helmet and Display Sight System, the Pilot Night Vision Sensor, the Target Acquisition and Designation System, the laser tracker/designator, and the image auto tracker. It provides the sensor interface between the crewmember and the environment. What follows is a brief description of the actual operation.

The primary function of the helmet sighting system is to increase the combat effectiveness of the helicopter by employing the excellent wide-angle discrimination, acquisition, and high-response features of the human crewmember's brain and eye to direct high-resolution, narrow-field-of-view target acquisition sensors to a target via his line-of-sight (LOS). As it is configured in the AH-64, it is a two-place system; that is, both the pilot and copilot gunner have helmet sighting systems, thereby allowing the pilot to direct the copilot LOS to a target of opportunity or threat with minimal acquisition time.

Many hardware components of the helmet display system are similar in both crew stations. Each crew member has his own helmet and display unit as well as electronics units. The LOS is determined by the use of two photodetectors on the helmet which are aligned parallel with the user's LOS. Sharp-edged IR fans sweep through the cockpit and trigger signal pulses in these detectors. These pulses, along with a reference detector, provide sufficient information to resolve, by trigonometric means in the electronics unit, the required azimuth and elevation angles of the wearer's LOS.

The pilot uses the night vision sensor for externally aided vision at night or in adverse weather conditions. It consists of a stabilized forward-looking infrared sensor contained in a rotating turret mounted at the front of the AH-64 aircraft. When selected, the night vision turret is slaved to the crewmember's LOS using the helmet sighting system.

The Target Acquisition and Display System provides the copilot with day/night, adverse weather target acquisition, giving him the capability to carry out missions that are impossible for other attack helicopters. This system provides a means to find targets, recognize them, and accurately determine relative position and range. Primary control is through the optical relay tube located in the copilot's cockpit.

The target acquisition system is made up of the following subsystems:

Forward-Looking Infrared—Provides a thermal image for use in reduced visibility or at night.

Direct View Optics—Provides a low/high-magnification telescope.

Day Television—Provides high and zoom magnification of a black and white TV video. This is used in poor visibility conditions and is very useful in finding targets in camouflage. It is also effective in maximum-range target acquisition.

Laser Rangefinder/Designator—Provides target ranging and precision laser-beam pointing. It provides the laser energy for autonomous designation delivery of the Hellfire missile.

Laser Spot Tracker—Provides the capability to acquire targets which are being lased remotely. The spot tracker slews the sensor LOS to the target at lock-on.

Image Auto Tracker—Allows video lock-on of contrasting images.

Integrated Helmet and Display Sight System (IHADSS) Simulation

Since the effectiveness of the training device for weapon delivery accuracy is affected by its component parts, the CMS employs much of the actual IHADSS hardware without modification. Simpler approaches such as an electromechanical simulation using a linkage attached to the helmet would be unacceptable in providing the exact pilot head movement and sensor slew correlation necessary for night vision flight.

Simulator-unique hardware includes serial data interface to the sight electronics, linkage for digital conversion, and

the MUX bus interface device which interfaces to the fire control computer via a 1553 bus. Controls include power and boresight commands from the cockpit panels.

The IHADSS is available for training in independent or integrated modes. In independent mode, the missing crewmember is replaced with a "missing man" function in software. This requires each simulator cockpit to have a complete set of IHADSS hardware.

The "missing man" function in independent mode includes status and avionics fault information as well as LOS data for the missing crewmember. Status/fail information is a mirror of the actual crewmember. LOS is set to 0 degrees azimuth and elevation unless an instructor-designated target is within the IHADSS field of view. In that case the LOS of the missing man will be toward that target.

Pilot Night Vision System (PNVS) Simulation

The simulation of the night vision system is done in the software of the host computer. It consists of mode control and the servo simulation. Inputs are from the fire control computer via 1553 bus as well as simulated air vehicle inputs and cockpit switches. The azimuth and elevation angles and rates are sent to the visual system for display line of sight by high-speed block transfer in the channel manager.

The primary goal of this simulation was to develop equations that described the elevation and azimuth servos. It was necessary that linear and nonlinear response characteristics, as well as aircraft-induced disturbances, be included.

To understand the process of simulating the analog servos incorporated in the night vision system, one must first know something of the actual servos used in the AH-64. There are two independently operating servos that together position the infrared receivers at some AZ and EL angular position with respect to the aircraft. One moves a circular platform, or gimbal, about a central pivot in the aircraft x-y plane and is called the AZ servo. The other is attached to the AZ gimbal and rotates the lens and mirror of the infrared sensor's optical system in a plane perpendicular to the aircraft x-y plane and is known as the EL servo. The travel limits of the servos are -120 to +90 degrees in AZ and +10 to -22.5 degrees in EL. The field of regard for AZ is limited from -90 to +90 degrees. For EL, the line of sight of the sensor is twice the mirror deflection, making the field of regard in EL +20 to -45 degrees.

The system utilizes rate servos. Rate commands are received from the aircraft TADS electronic unit in the form of a voltage analog, and the inertial rate of the gimbal, as measured by a gyro, is compared with the desired rate to produce the AZ and EL servo rate error. The servo rate error is operated on by the servo compensation electronics and the prime mover to produce a torque about the motor armature. The motor and gimbal rate continues to vary until the servo rate error produces enough torque to overcome system coulomb friction and maintain the desired rate. Mounted on the gimbal is a resolver used to measure the instantaneous angular position of the gimbal relative to the aircraft body. The resolver output is then sent back to the electronics unit via resolver-to-digital converters to be used in the computation of the next rate command to the servos.

In order to digitally simulate the night vision servos in real time, the math model describing the analog system must be band-limited to eliminate the higher natural frequencies in the system. This is important because the higher the frequencies that must be simulated, the smaller the quadrature interval used in the numerical integration algorithm becomes, and the higher the number of passes through the program needed to compute real-time information. Band-limiting the system is not restrictive from a high fidelity simulation standpoint when the frequency limit is higher than the servo's closed-loop bandwidth.

An equivalent system is developed by lumping the total system inertia at the point where the rate feedback is measured. For AZ, the total inertia is lumped at the gimbal; for EL, at the lens. All torques (friction and nonfriction) on the equivalent inertia are then computed and summed. Dividing the resultant torque by the equivalent inertia produces the inertial acceleration for the current pass. Subtracting the current aircraft acceleration and multiplying by the band-drive gear ratio for EL results in the AZ gimbal and EL mirror acceleration referenced to the aircraft. The inertial acceleration is integrated using a second-order Adams-Bashforth numerical integration algorithm to produce the inertial rate for the next pass. Finally, the rates of the AZ gimbal and EL mirror referenced to the aircraft body are integrated to obtain the angle for the next pass.

Various saturation and limit nonlinearities are accounted for in the process. These include mechanical travel limits on the gimbal and mirror, maximum torque and rate limits on the prime movers, and voltage limits on the inputs and servo error.

This simulation approach was verified against a model using exact coding of the real system block diagram. The model was run off line at a high rate. The simulated version compared very well with this and gave assurance of the high-fidelity simulator servo response necessary for night flying training. This is particularly true with rapid head scanning.

Target Acquisition and Designation System (TADS) Simulation

The TADS simulation consists of a software emulation of the TADS electronics unit and a simulation of the TADS turret. It provides the visual system, through the channel manager, the TADS turret line of sight for display purposes. Inputs are from the cockpit, image auto tracker, and fire control computer.

The electronics unit simulation determines power and moding. Control rates can be supplied from the copilot manual thumb-force, linear motion compensation, the image tracker hardware, or the laser tracker.

The manual thumb-force from the copilot input supplies azimuth and elevation rate commands. Initially these commands are represented by voltages. A nonlinear conversion transforms them to rates for the turret simulation. These rates are then scaled based on the types of sensor and field of view selected.

If linear motion compensation is selected, azimuth and elevation compensation rates are added to compensate for variations due to aircraft translational motion relative to a particular target or scene located at a certain range. Variations in target coordinates due to helicopter position changes or target motion are balanced out by subtracting the effective change in TADS azimuth and elevation angle to provide for the TADS position cancellation. Since the commanded rate is being integrated during motion compensation, the current rate will be held constant if the copilot removes the pressure on the thumb-force controller.

The image tracker is a balanced area tracker. The image tracker system contains all the necessary hardware and error processing algorithms to provide azimuth and elevation error signals proportional to the displacement of a target image from the center of the television or infrared field of view (FOV). These errors are scaled, based on selected sensor and field of view, filtered, and used to compute AZ and EL rates for turret dynamics. The image tracker simulation is discussed later. If the copilot selects offset mode, the thumb-force AZ and EL control inputs are limited and used to offset the currently tracked image from the center of the selected sensor display.

For the image tracker mode, missile obscuration effects must be taken into account. If a missile has been launched from the ownship, the gimbal rates are momentarily frozen so the image tracker does not break lock due to the missile blast brightness.

The laser tracker in auto mode requires a four-bar scan pattern. This pattern extends ±30 degrees horizontally and +5 degrees to -25 degrees vertically about the initial LOS in pitch and roll stabilized coordinates. During each iteration an LOS command is computed and converted to rate commands.

The TADS turret dynamics computes TADS earth-referenced LOS angles to be used by the visual system and as a feedback signal to control rate software in the TADS electronic unit simulation. Simulated TADS platform rates and accelerations are also computed for use by the visual system in its extrapolation routine. Rate commands are received and integrated to yield the desired azimuth and elevation angles. For manual track and image tracker modes, stabilization rates are also computed. The stabilization approach is based on a first-order loop control of the TADS azimuth and elevation angles with aircraft pitch and yaw perturbations. Variations in the earth-to-body matrix, received from the air vehicle simulation, are used to compute a corresponding change in the TADS-to-body matrix to null the helicopter attitude perturbation; thus, changes in the helicopter pitch and yaw parameters will not affect the TADS earth LOS if within the gimbal limits. These stabilization rates are then added to the commanded rates to obtain the overall change in earth-referenced LOS. No stabilization is computed for aircraft roll.

Laser Rangefinder/Designator Simulation

Because the flight of the Hellfire missile was so rigorously simulated (as will be discussed below), it was necessary to provide a laser designation simulation of comparable complexity. This includes the divergence of the beam and visibility effects.

The priority sector processor is continually loaded for range requests. These requests give an origin point of the laser rangefinder and an attitude matrix of the LOS vector for processing. In order to simulate beam divergence, multiple requests slightly off the sensor LOS were used. Processing was optimized in software to ensure maximum range request frequency as required for the simulation. The return from the priority sector processor is the range along the requested LOS as well as the intersection type (i.e., target or ground).

The FCC uses a Kalman filter for determining final range for weapon delivery. If the range requests are at too low a frequency it is possible for the filter to de-correlate and go into a coast mode, resulting in poor weapon accuracy.

Extrapolation of the data sent to the priority sector processor was necessary to provide accuracy for moving targets. Because of the processing delay, lead compensation was necessary to have the beam correlate with the visual display. The extrapolation used both rate and acceleration of the TADS LOS.

Laser Spot Tracker Simulation

The laser spot tracker is part of the overall remote designation simulation. This allows the instructor to become a remote designator. He has control over designation, target

code, and power on. The spot tracker's LOS can be slewed via the TADS in a manual or auto scan mode.

Training requirements for the tracker included anomalies in lock-on due to visibility and remote designator location as well as code select. Rigorous signal models were necessary to meet this requirement.

Image Auto Tracker Simulation (IAT)

The IAT simulation represented one of the highest-risk areas. The tracker has many anomalies in its target-tracking operation which depend on the gain/level setting, weapon effects near the targets, and trees or other terrain features near the targets. In order to get these important effects it was determined that a part of the actual IAT hardware would be used rather than attempt a software simulation. This hardware included video tracker, video processor, and timing hardware.

The entire system is closed-loop, originating with visual video inputs through the video tracker. This generates error signals which then interface through analog/digital conversion to drive the TADS servo LOS simulation for the visual system, completing the loop. Analysis using off-line system modeling early in the development indicated a marginally stable system, which caused concern over the approach. Classical lead-lag compensation increased the phase margin to make the system stable and maintain tracker response.

It was found that the simulated image tracker performed well on the computer-generated imagery.

Armament

Weapon Types

The AH-64 Apache attack helicopter is armed with Hellfire missiles, 2.75-inch folding-fin rockets, and a 30-mm gun. Either the pilot or the copilot/gunner can fire any of the ordnance, two different weapons being selectable at any time, although the Hellfire or rockets may require cooperative effort by both crewmembers.

Flight paths are computed for all weapons using standard equations of motion with real-weapon-derived aerodynamic coefficients. Simulator iteration rates for these models are high enough to produce stable solutions with reasonable accuracy, while attempting to minimize software execution times.

Weapon Modeling

The choice of a full ballistic equation-of-motion model is expensive, both in software generation cost and in computer processor time. For a full mission CMS, however, the cost is more than justified by the training achieved, training which is unobtainable outside the simulator. Flying in nap-of-the-earth conditions and launching weapons from masked locations requires developing special crew skills. Terrain clearance must be assured or a weapon such as the Hellfire could impact the ground in front of the Apache, with loss of an expensive missile. Rockets interact with rotor downwash and wind conditions, and, along with the gun, require "Kentucky windage" corrections to be made while flying in stressful situations at or below treetop level. As a result, ballistics equations were mechanized for all weapon rounds.

Hellfire

The Hellfire is a laser-guided missile, homing on laser energy scattered from the target. Laser energy originates from a designation device which can be either a man-portable ground laser locator-designator or an airborne unit such as the laser rangefinder/designator, a component of the Apache target acquisition and designation sight system. Since the Hellfire is a guided missile, the modeling is complex, requiring emulations of the seeker system and autopilot, as well as the aerodynamic equations of motion. Other simulation software takes designation data and, interacting with the visual database, determines where the designator line of sight intersects an object. This position is the designate point used by the missile guidance equations.

All on-board processors that interface with the Hellfire are emulated in the CMS, providing on-the-rail and launch control exactly as in the real Apache.

Missile Ballistics. A five-degree-of-freedom (DOF) model is used for the Hellfire missile, the roll control being assumed perfect. Standard aerodynamic equations are used to compute forces and moments on the missile, and hence the missile position and attitude. Iteration rates must be high (by training simulator standards) in order to get a stable model. This high rate is necessary, in part, because of the presence of the seeker and autopilot closed loop.

As the missile iterates out in position, the flight path is checked for intersection with the visual database. Thus the actual impact point can be determined for scoring routines and for positioning visual weapon bursts. Missile position and attitude are used to drive a visual representation of the Hellfire departing from the Apache and eventually homing in on the designated point.

Aerodynamic coefficients used in the model were derived from wind tunnel data, and comparisons between the simulation and actual flight tests show good fidelity in the CMS model.

Missile Seeker. The missile seeker line of sight and field of view are simulated along with angular and angular-rate limits. The line of sight is checked in the visual database to determine if any obscuration is preventing the seeker from

actually "seeing" the returning laser energy. If no obscuration is present, guidance errors are passed to the autopilot.

Seeker search patterns and detection sensitivity emulate the real world for both on-rail and in-flight conditions, maintaining the high fidelity of the Hellfire model. Since the missile/designation point geometry is relatively slow-changing, the seeker model runs at a lower rate than missile ballistics.

Autopilot. The autopilot model emulates the real missile transfer function with the exception of the roll channel. Seeker guidance errors are used to generate fin position drives which in turn modify the aerodynamics, and hence the missile attitude and position.

Rocket Simulation

Five different types of rockets are included in the CMS weapon loads. There are two kinds of high-explosive rounds, an illumination round, a smoke round, and a multipurpose submunition round. All types are unguided.

The aerial rocket control system (ARCS) processor, which in the Apache controls the selection, fusing, and launch of the rockets, is fully emulated in the CMS, the ARCS high-level source language being translated into Fortran in the simulation host computer. Availability of the source software considerably simplified this effort.

Rocket Ballistics

The rocket ballistics model is the same 5-DOF model used for the Hellfire, but without seeker or autopilot. Aerodynamic coefficients used are appropriate to each type of rocket. The lack of an autopilot-seeker system allows an iteration rate lower than the missile to keep a stable model. Further economies in computation resources are realized by treating the firing of a pair or quad of rockets as a single trajectory calculation. Individual rocket positions needed for visual or scoring purposes are obtained by applying simple offsets to the computed trajectory.

As mentioned before, rotor downwash has a marked effect on the rocket, which is relatively slow moving as it passes through the downwash area. Complete modeling of the downwash would be difficult and expensive, so the CMS uses an average downwash value, computed for the current flight conditions. This approach provides sufficient fidelity as verified by comparison with flight test data.

The only visual representation of a rocket is a light point (the rocket motor), which is visible until motor burnout, and the effect at impact (explosion, drifting flare, smoke, or submunition pattern).

Gun Simulation

The Apache 30-mm gun is a turret-mounted unit, the turret being slaved to the selected sighting device. This can be either the helmet sight of the pilot or gunner, or the TADS system. The on-board fire control system applies corrections to the gun position to account for engagement range, ownship motion, etc.

The CMS includes a mathematical model of the gun turret, ensuring correct rate responses and angular limits. This fidelity is further enhanced by using a stimulated aircraft fire control computer (FCC), with unmodified aircraft software. Sighting inputs are thus processed in a manner identical to that in the Apache helicopter.

Gun Ballistics

Since the in-flight gun round is a symmetric spinning object, the ballistic equations can be simplified to a three-degree-of-freedom model with corrections. These corrections account for the gyroscopic effects of the spinning bullet moving in a gravitational field and are based on flight-test data.

Trajectory calculations are made for only one of a maximum of five rounds, the position of the other rounds in that group being accounted for in the size of the dispersion pattern about the calculated impact point. This allows economy in computer resources.

Scoring

The weapon impact positions are used to compute how successful the student was in hitting his target. Data is fed to the instructor, by way of a graphics display page, showing what was engaged, ownship conditions at weapon firing time, range to the target, and outcome of the engagement (hit, miss, or kill). Determination of the target being attacked is based on which target is being designated at the time of weapon impact (for the Hellfire missile), or by which target is closest to the sighted point (for the rockets or gun).

Each target type has a vulnerability associated with it for each weapon type. Thus a hit with a high-explosive rocket on a particular target type might give a kill, while a hit on a different target would only score as a hit (no kill). Killed targets cease to be active in the mission scenario, displaying a permanent smoke column. Targets which have been hit, but not killed, will show a weapon burst followed by a dissipating smoke cloud, and will continue to interact with the Apache, launching weapons at the Apache where appropriate. Misses produce visual effects similar to the non-kill effect, but not necessarily on a target.

Environment and Survivability

Gaming Area

The Apache CMS visual gaming area is a 32-km-by-40-km region of generic terrain designed to fulfill the combat mission training requirement. Development of such a large wholly unique area of database would have been prohibitively expensive, so the CMS used a "block" approach in which 1-km-square blocks are put together in a mosiac fashion.

Database Blocks

The gaming area is broken up into 1-km-square blocks, each block being chosen from a library of available types. The blocks can be oriented in any of four directions and are designed to connect to adjacent blocks by allowing only a limited number of edge terrain profiles. In this way, a specific block type may appear at several different positions in the gaming area. Further economies in database design are realized by the "derivative" blocks which retain the same terrain features, but add or delete road segments or buildings. Even though terrain features are repeated, the database is sufficiently large so that this is not evident to the student aircrew.

Threat Simulation

The heart of the AH-64 CMS tactics simulation is the semi-intelligent threat[1]. The student enters a scenario in which there exist not just targets, but realistic, weapon-carrying threats. Routes must be chosen with care and exposure minimized or else hits will be taken. Hits or near misses produce motion, aural, and visual cues with sufficient realism to instill considerable respect for the threat and rapidly teach correct flying techniques.

Ten threats can be present at any time, but as the mission develops, threats which, as a result of ownship movement, for example, are no longer tactically significant, are automatically removed and new threats take their place. Threats can be placed at any of 99 sites, 20 of which have predefined pathways. As many as five moving threats may occupy a single moving target site. Non-friendly vehicles with weapon capability may or may not be hostile. Hostility and movement (for threats able to be hostile or threats on moving target sites) can be preprogrammed to occur as a function of time, ownship position, or other factors. For example, a threat might become hostile and start engaging the ownship following use of the Apache weapons against a target.

Threat Types

Sixteen different types of vehicle can be assigned to a target site. These are the T-62, T-80, and M-1 tanks; SA-4, SA-6, SA-8 (radar and optically guided), and SA-9 surface-to-air missiles; ZSU-23-4 gun; BMP gun; BTR-60 troop carrier; heavy and light trucks; Flapwheel radar; and the Mi-2 scout and Mi-24 Hind helicopters. Current CMS design limits hostility to the T-62 and T-80 tanks, the missiles, and the ZSU and BMP guns. Threat updates being investigated currently will expand the CMS into the air-to-air role, requiring an interactive air threat.

Threat Movement

Twenty of the 99 target sites support moving targets, five of the 20 being dedicated to air targets. Each site has one to five pathways, a single target occupying a given pathway at any time. The pathways are predefined routes through the database, each made up of a series of linear segments which, for the ground paths, follow the profile of the terrain.

Target speed along the path is specified when the mission scenario is generated, but the target may be inhibited from moving until some predetermined conditions are met. Thus the target could lie dormant until a particular mission phase or ownship position occurred, at which time the target would accelerate to its predefined speed. Moving threats will also respond to the intelligent lethality algorithm, decelerating to a stop prior to aiming weapons at the Apache ownship, or accelerating away when weapons are depleted. Target speed also varies as a function of the terrain slope over which it is moving, slowing down as it climbs hills and speeding up on downslopes. These variations contribute to the establishment of a realistic workload on the student during target tracking and weapon delivery.

Threat Intelligence

The AH-64 CMS threat is controlled by a semi-intelligent "lethality" algorithm, developed to provide a realistic adversary in all phases of combat mission training. The algorithm calculates probabilities of acquisition and hit based on the current situation and the threat capability.

Probability of acquisition is determined from three basic parameters, each of which bears equal weight:

- The range from the ownship to the threat
- The length of time the ownship has been exposed to the threat
- The angular height of the ownship above masking terrain

Threat capability for each threat type is stored in the threat library, specially organized areas of memory that provide easy threat update since all parameters for a given threat are located in one place. Given the range to the threat, the exposure time, the height above mask, and the threat capability, the probability of acquisition can be determined. The probability of acquisition is then modified as a func-

tion of time of day (day/night), visibility (as selected by the instructor), and use of airborne survivability equipment. The instructor specifies a "lethality level" or threat skill level which is used to determine a threshold for acquisition. When the probability of acquisition exceeds this threshold, the threat is considered to have acquired the Apache and threat weapon aiming and subsequent weapon firing will occur.

Threat aiming and weapon fire or missile launch are shown on the student visual displays. Following appropriate weapon time of flight, the probability of hit is computed. This also uses threat capability, threat range, height above mask, and survivability equipment utilization data, and compares the result with the probability-of-hit threshold. This threshold is based on the lethality level, in similar fashion to the probability of acquisition. If the probability of hit exceeds the hit threshold, a weapon hit is considered to have occurred; otherwise a "near miss" is scored. Hits or near misses produce student visual system and motion system responses, the hit momentarily turning the displayed scene red, and jarring the cockpit, while a near miss shows the weapon burst and provides a lower-amplitude jolt. Both effects also include appropriate aural responses.

A hit can trigger prespecified malfunctions, determined for each threat at mission preparation time. They can range from nuisance effects like popped circuit breakers to catastrophic effects like the loss of engines or tail rotor. Thus the student must react to the hit, either by taking cover and continuing to fight, or by attempting a crash landing.

Scoring the Student vs. the Threat

As the student moves through his mission and is acquired and attacked by various threats, data is passed to the instructor by way of a graphics page. For each engagement, the threat type, location, exposure time, range, mask utilization, acquisition probability (maximum, mean, and current), and outcome (hit or miss) are shown. The instructor can review this data and provide immediate constructive criticism, or make a hardcopy of the page for use in the debrief session following the mission.

Lethality level is also included on the threat scoring page and thus review of the hardcopies for a given student can show his progress over a period of time, or as the threat increases in skill.

Airborne Survivability Equipment (ASE)

The AH-64 CMS includes full simulation of all the Apache ASE, including the interaction of the on-board systems with the CMS environment. The APR-39(V) 1 radar warning receiver responds to the presence of radar threats, showing the correct strobes on the display and providing realistic audio to the aircrew. The ALQ-136 and ALQ-144 jammers affect threat operation as appropriate, causing corresponding changes to acquisition and hit probabilities and thus changing the threat weapon hit or miss result. Use of the M-130 chaff dispenser also modifies threat operation and the resulting score.

Radar Warning

Radar warning cues are often the first warning the Apache crew has that threats are active in the area. The CMS provides these cues using a stimulation of the APR-39 system at the video level. In the real world, the receivers output video pulse trains to the processor for analysis and display. Specialized simulation hardware is used in the CMS to generate these pulses, which are then used to stimulate an actual aircraft processor. Cockpit displays also use real-world equipment, giving an APR-39 simulation with very high fidelity.

Threat radar parameters are stored in the threat library, just like the other threat parameters, and thus, since all information is located at one place, the CMS threat complement can be changed or updated with relative ease.

Jamming and Chaff

On-board jamming equipment is simulated inasmuch as the student can interact with or monitor the systems, and also in the effects the jammers have on the threat environment. The student interaction or monitoring requirements are relatively simple—provision of switches and lights and simulation of the self-test functions Threat interactions are somewhat more complex, requiring analysis of jammer capability against specific threats and threat weapons. The overall result is an integral part of the threat lethality model providing a realistic and responsive battle environment for the aircrew.

Chaff is also simulated from the same two points of view. Cockpit switchology is identical to that in the real Apache, while threat interaction with the deployed chaff is determined by the threat parameters. Simulated chaff clouds dissipate over a period of time and thus decrease in effectiveness.

The capability of chaff or jammmers against a specific threat is part of the threat library, again ensuring ease of update, as well as consistent threat operation.

Current Status and Future Enhancements

Currently two production units of the CMS are in use for training. Student and instructor comments on the CMS have been outstanding. The high-fidelity simulation described in this paper has made the CMS a total mission simulator.

Numerous CMS enhancements are being considered by the Army, including such tactics-related upgrades as incorporation of -37 fire control computer software. In addition,

there is much excitement over the prospect of pursuing the challenges of a whole new dimension of helicopter combat mission training. The new dimension is air-to-air and the challenge will be development of the simulation necessary to support training in air-to-air tasks, the feasibility of which has been demonstrated recently with the CMS. In addition to being a highly effective training device, this simulator could be also instrumental in the development of air-to-air helicopter tactics in a total mission scenario.

References

1. Yarlett, W. A., Drew, E., "Combat Mission Simulation: The Attractive Alternative." Published Proceedings of Interservice/Industry Training Systems Conference, Salt Lake City, Utah, Nov, 1986.
2. Yarlett, W. A., "Combat Mission Simulator—Kill or Be Killed." (three parts) United States Army Aviation Digest, March, April, May 1985.